



LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-TR-630332

XML Network Description Language (XNDL) for ns-3

ns-3 XSD v0.9

L. E. Banks, P.D. Barnes, Jr., D.R.
Jefferson, S. Nikolaev
February 2013

Technical Report
submitted to the
United States Army Research Lab (ARL)

Disclaimer

This document was prepared as an account of work sponsored by an agency of the United States government. Neither the United States government nor Lawrence Livermore National Security, LLC, nor any of their employees makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights.

Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States government or Lawrence Livermore National Security, LLC. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States government or Lawrence Livermore National Security, LLC, and shall not be used for advertising or product endorsement purposes.

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Table of Contents

Introduction	2
1 Overview	3
2 Document Conventions and History.....	3
2.1 General.....	3
2.2 Fonts.....	3
2.3 Terminology	3
2.4 Revision History.....	4
3 Interfacing With ns-3	4
3.1 XSD Root Element: <NetSim...>.....	4
4 XNDL Top-Level Elements	5
4.1 <NodeContainer...>	5
4.2 <Subnet...>.....	5
4.3 <Application...> and <ApplicationSet...>	6
4.4 <MPIEnabled...>.....	8
5 Support for High Performance Computing and MPI	8
6 Ongoing and Future XNDL Development.....	10
7 Bibliography	12
Appendix A. XNDL for ns-3 v0.9 – XSD Reference	A-1

Introduction

This document is the “XML Network Description Language (XNDL) for ns-3 ns-3 XSD v0.9” addressing the Statement of Work for the Proposed Research in Network Simulation under contract L145271. This work performed by Lawrence Livermore National Laboratory (LLNL) is specifically for the Army Research Laboratory.

There exists a dichotomy between a desire to create generic network models that are framework-agnostic and models that are tailored to specific simulation implementations. Both methods have advantages and disadvantages. A generic approach would yield models that can be used and shared among many implementations and thus potentially gain support of a larger user community. However, there would remain a gap between concepts in a generic model and the corresponding representation needed for specific implementations. An approach that targets a specific implementation allows models to take full advantage of tighter integration and have a more explicit representation of simulation concepts and relationships. But locking into a specific implementation removes the portability aspect of the model. See Barnes, *et al*, for a more complete discussion [1].

XNDL is a XML-based architecture being developed at LLNL for the purpose modeling networks at a high level of abstraction while using a hierarchical structure of targeted grammars to interface with a broader base of network simulation frameworks and their associated tools (see Figure 1).

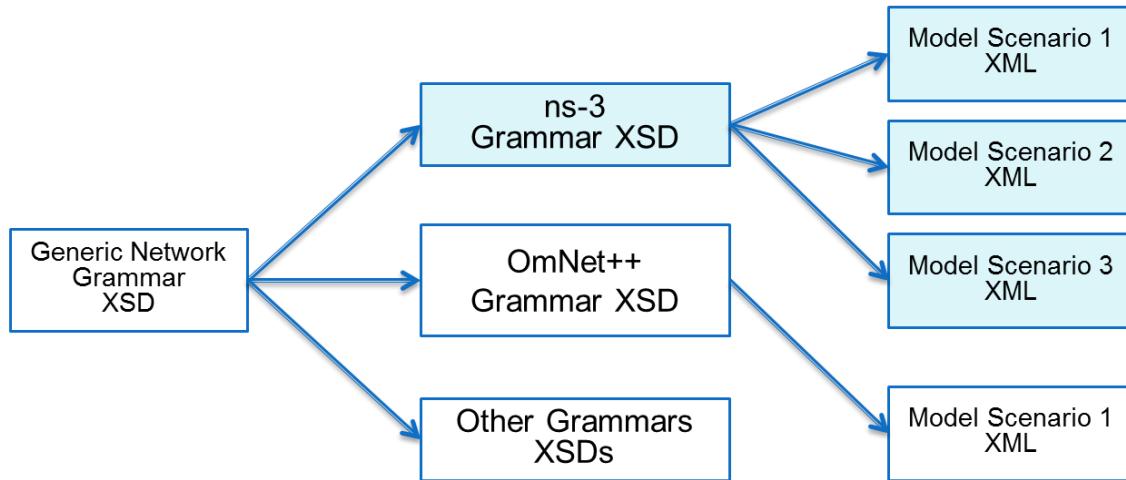


Figure 1: XNDL hierarchical architecture. XNDL for ns-3 is highlighted.

This document describes *XNDL for ns-3* including the grammar defined with a XML Schema Definition, or XSD, and the coupling between generated model scenarios and the ns-3 simulator. Supporting simulations on high performance computing clusters using MPI is also discussed. The full XSD for the ns-3 grammar is included in Appendix A.

1 Overview

When working with a simulation tool there is often a desire to abstract the model from the simulation. This abstraction enables a modeler to focus on the objects, behaviors, and relationships being modeled, independently of the underlying implementation. A model definition can be retained and versioned, enabling repeatable simulations and tightly controlled studies by varying specific parameters in the model. If differing implementations of the simulation exist, an abstracted model can be more easily ported or *mapped* from one simulation framework to another.

The ns-3 Network Simulator lends itself nicely to this idea of creating a network model that is only loosely coupled to the actual implementation. Out of the box, ns-3 incorporates user defined *scripts*, which are actually C++ or Python programs, to instantiate the objects with the desired parameterization. It is straightforward to have a parameterized model that can be parsed and interpreted to instantiate the corresponding objects in the simulation, by using the same mechanism as the ns-3 scripts.

XML has been chosen as the language to represent XNDL. XML is a popular format for expressing models with well-defined entities and relationships in an intuitive manner. An object in the real world can be represented by an XML element with a given set of attributes to capture the desired aspects of the object being modeled. Complexities of objects can be decomposed and represented by parent-child relationships in XML or composition of multiple XML elements. An XML Schema Definition, or XSD, specifies the *grammar* to which an XML representation must adhere.

2 Document Conventions and History

2.1 General



Notes are indicated by the clipboard icon to the left of text. These are used to make comments regarding planned future modifications and extensions.

2.2 Fonts

ITALICS	Used to emphasize a new concept, term, or definition
BOLD	Used to emphasize a specific section of an XML code examples and snippets
FIXED WIDTH	Used for all XML code examples and tags.

2.3 Terminology

An XML *tag* will refer to either an *opening tag* for an element: <element>, or a *closing tag*: </element>. An XML *element* will refer to the opening and closing tags and all information contained within.

2.4 Revision History

The current version of XNDL for ns-3 is 0.9. The following table will track the major modifications of this document by date in relation to the progressing versions of XNDL for ns-3.

Document Revisions

XNDL Version	Date	Contact	Remarks
0.9	2/22/2013	Eddy Banks	Initial version.

3 Interfacing With ns-3

The current XSD allows specification of connected P2P and CSMA network topologies, and support for the configuration and distribution of several ns-3 applications. It also supports the configuration of packet capture (pcap), and trace file generation. The ns-3 simulator implementation has been augmented with a front-end module called: XMLSimulationReader that parses an input XML model and instantiates the ns-3 simulation objects using ns-3 helper classes in the same way as a traditional user-created script.

3.1 XSD Root Element: <NetSim...>

The root element of the XSD model is <NetSim>. The NetSim element contains attributes needed to initialize the simulation as well as containing the current schema version. The attributes are as follows:

- Name (required)
 - User-defined string name for this simulation model.
- P2pEnablePcapAll
 - Boolean flag to turn on pcap format tracing for point-to-point devices.
- P2pEnableAsciiTraceAll
 - Boolean flag to turn on ascii tracing for point-to-point devices.
- CsmaEnablePcapAll
 - Boolean flag to turn on pcap format tracing for CSMA devices.
- CsmaEnableAsciiTraceAll
 - Boolean flag to turn on ascii tracing for CSMA devices.
- SchemaVersion (required)
 - Current version of XNDL ns-3 model.

Retaining the schema version is important in order to record what version of the model was used with a particular study. It is also needed as a connection point for the various tools in the XNDL framework. For example, various transformation tools, and the simulation implementation itself, may only be applicable to a specific version of the model and may not be applicable for older or newer versions.

The ns-3 network related elements are children of the `<NetSim>` root. These top-level elements are described next.

4 XNDL Top-Level Elements

4.1 <NodeContainer...>

The `<NodeContainer>` element in XNDL for ns-3 has a `Name` attribute and an optional `Size` attribute. If `Size` is given, then `Size` number of ns-3 nodes will be added to the model. The `Name` attribute is used to define a unique name allowing other elements to refer to a specific `<NodeContainer>`. If the `<NodeContainer>` element does not specify the `size` attribute, then it is expected to contain `<RefNode>` and/or `<ApplicationSet>` child elements. Each `<RefNode>` contains a `name` attribute referring to another `<NodeContainer>` containing the actual node, and an `index` attribute referring to the index of the node in the referenced `<NodeContainer>`. For example, in Snippet 1, the `<NodeContainer>` with `Size` attribute has `Name="ALL_NODES"`. Other `<NodeContainer>` elements contain `<RefNode>` children that refer to the nodes that were created in the "ALL_NODES" `<NodeContainer>`.

```
. . .
<NodeContainer Size="10" Name="ALL_NODES"/>
<NodeContainer Name="csma_1_nodes">
    <RefNode Name="ALL_NODES" Index="0"/>
. . .
```

Snippet 1: `<NodeContainer>` referencing.

4.2 <Subnet...>

The `<Subnet>` elements contain attributes needed to parameterize the network linkage between nodes. As seen in Snippet 2, the `<NodeContainer>` is referenced by `Name` and again the actual nodes are linked via the `Index` attribute on a `<RefNode>` child element.

```

. . .
<Subnet Cidr="10.1.1.0/24" Type="CSMA" Name="csma_1"
NodeContainer="csma_1_nodes" DataRate="100Mbps" Delay="7ms">
    <Description>csma_1_Subnet</Description>
    <RefNode Type="ROUTER" DnsName="router1.llnl.gov" Index="0">
        <IPAddress>10.1.1.1</IPAddress>
    </RefNode>
    <RefNode Type="SIMPLE" DnsName="node6.llnl.gov" Index="1">
        <IPAddress>10.1.1.2</IPAddress>
    </RefNode>
</Subnet>
. . .

```

Snippet 2: <Subnet> element

In this case, the `Index` attribute of the `<RefNode>` element refers implicitly to the `<NodeContainer>` attribute of the containing `<Subnet>` element. As can be seen, the `<Subnet>` element contains enough information to invoke several ns-3 helper classes.

4.3 <Application...> and <ApplicationSet...>

XNDL for ns-3 contains models for several of the ns-3 example applications including:

- bulkSend / PacketSink
- csmaPing
- genericApp
- onOffApp
- udpClientServer
- udpEcho

The `<Application>` element contains a user-defined `Name` attribute as well as any attributes specific to the application. The `<ApplicationSet>` element contains `<Application>`s and other `<ApplicationSets>` incorporated by reference to their `Name` attributes. In turn, a `<NodeContainer>` can refer to `<Application>`s and `<ApplicationSets>` by reference to the same `Name`, which has the effect of instantiating the referenced applications on nodes in the `<NodeContainer>`. As seen in Snippet 3, the `<NodeContainer>` installs two different `<ApplicationSets>` on specific nodes.

```

. . .
<NodeContainer Name="csma_4_nodes">
    <RefNode Name="ALL_NODES" Index="3"/>
    <RefNode Name="ALL_NODES" Index="8"/>
    <ApplicationSet Name="WebBrowsingSet0_0" Index="0"/>
    <ApplicationSet Name="WebBrowsingSet0_1" Index="1"/>
</NodeContainer>
. . .
<Application xsi:type="OnOffAppType" Name="Server_1"
Protocol="ns3::TcpSocketFactory" DataRate="5000bps" PacketSize="500"
RemoteAddress="10.1.10.1" Port="80" Start="62.3564" Stop="200"/>

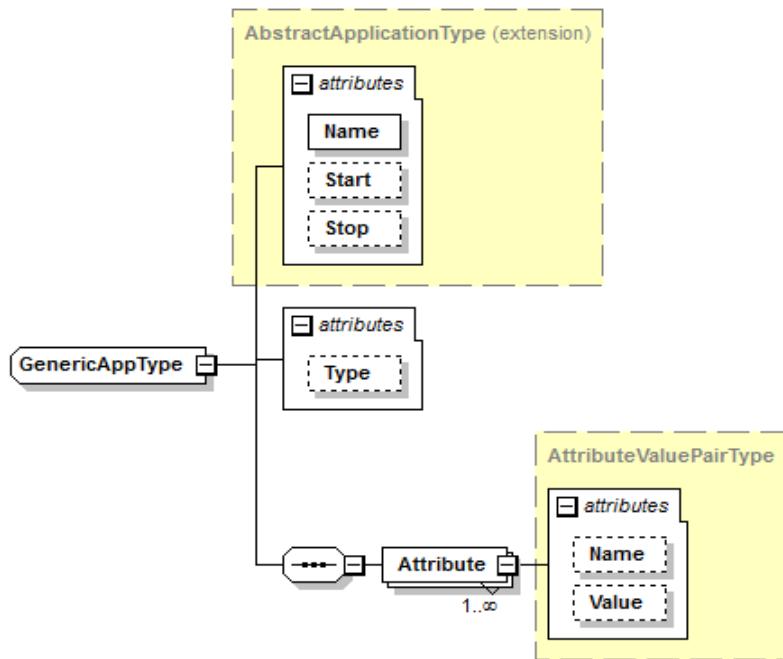
<ApplicationSet Name="WebBrowsingSet0_0">
    <Application Name="PacketSink"/>
    <Application Name="Client_0"/>
</ApplicationSet>
. . .

```

Snippet 3: `<Application>` and `<ApplicationSet>` elements.

4.3.1 Generic Application Element

XNDL provides support for applications which are not yet specified in the XSD. A generic application type, modeled with `xsi:type` of `GenericAppType`, has attributes for `Name`, `Start`, `Stop` and `Type` – but also contains `<Attribute>` child elements each of which contain `Name` and `Value` pair attributes to specify parameterization of the application. The XSD diagram for `<GenericAppType>` is shown in Figure 2.



Generated by XMLSpy

www.altova.com

Figure 2: `GenericAppType` diagram.

An example of defining a generic application with similar parameters as the OnOffAppType shown in Snippet 3 is shown in Snippet 4.

```
.
.
<Application xsi:type="GenericAppType" Type="MyOnOffAppType"
Name="Server_1" Start="62.3564" Stop="200">
    <Attribute Name="Protocol" Value="ns3::TcpSocketFactory"/>
    <Attribute Name="DataRate" Value="5000bps"/>
    <Attribute Name="PacketSize" Value="500"/>
    <Attribute Name="RemoteAddress" Value="10.1.10.1"/>
    <Attribute Name="Port" Value="80"/>
</Application>
.
.
```

Snippet 4: Defining a generic application.

As with the other application types, the parent `<NodeContainer>` specifies the Nodes on which the application will run via `<RefNode>` elements.

4.4 <MPIEnabled...>

This is a simple element that currently serves as a flag to indicate if the simulation is to be a parallel implementation running on a high performance computing cluster using message passing interface (MPI). More detailed discussion follows in section 5: Support for High Performance Computing and MPI.



Since this element is specific to the model as a whole it will be moved to an attribute of the `<NetSim>` root node in the next release of XNDL for ns-3.

5 Support for High Performance Computing and MPI

One of the goals at LLNL is to parallelize the network simulation using Message Passing Interface, MPI. To support this in the Model, the XSD includes an extra XML element: `<MPIEnabled>1</MPIEnabled>` to indicate a MPI job. The parameter value of 0 indicates a non-MPI simulation; any non-zero value indicates a parallel network simulation. This element is a child of the `<NetSim>` element as shown in Snippet 5.

```
<NetSim xmlns:xsi=http://www.w3.org/2001/XMLSchema-instance
xsi:noNamespaceSchemaLocation="NetSim.xsd" SchemaVersion="1.0"
Name="SimTest" CsmaEnableAsciiTraceAll="false"
CsmaEnablePcapAll="false"
P2pEnableAsciiTraceAll="false"
P2pEnablePcapAll="false">
    <MPIEnabled>1</MPIEnabled>
.
.
```

Snippet 5: Element to indicate MPI run.

In preparing a simulation input for a parallel run, the network topology must be partitioned and assigned to ranks for processing on the computing cluster. The process flow is shown in Figure 3.

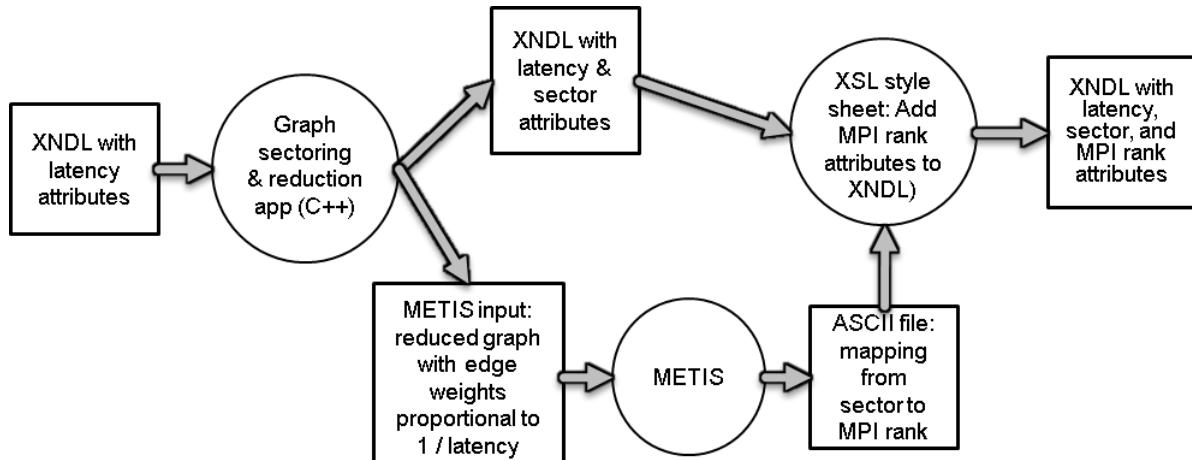


Figure 3: Automatic partitioning process flow.

The network model is first annotated for partitioning by the use of a `Sector` attribute on the `<NodeContainer>` and `<RefNode>` elements. All nodes with the same `Sector` value must end up on the same MPI rank, because the underlying channel cannot be partitioned. For example, as shown in Snippet 6, a `<NodeContainer>` whose nodes are all on a single CSMA channel should remain in the same partition, thus the `<NodeContainer>` is assigned a `Sector` attribute for that CSMA channel. A `<NodeContainer>` referencing nodes on a point-to-point channel may be partitioned, hence the `<RefNode>`s can have different `Sector` values.

```

. . .
<NodeContainer Name="csma_1_nodes" Sector="0">
  <RefNode Name="ALL_NODES" Index="0"/>
  <RefNode Name="ALL_NODES" Index="2"/>
  <ApplicationSet Name="WebBrowsingSet0_0" Index="0"/>
  <ApplicationSet Name="WebBrowsingSet0_1" Index="1"/>
</NodeContainer>
<NodeContainer Name="csma_2_nodes" Sector="1">
  <RefNode Name="ALL_NODES" Index="1"/>
  <RefNode Name="ALL_NODES" Index="3"/>
  <ApplicationSet Name="WebBrowsingSet1_0" Index="0"/>
  <ApplicationSet Name="WebBrowsingSet1_1" Index="1"/>
</NodeContainer>
<NodeContainer Name="p2p_1_nodes">
  <RefNode Name="ALL_NODES" Index="0" Sector="2"/>
  <RefNode Name="ALL_NODES" Index="1" Sector="3"/>
  <ApplicationSet Name="WebBrowsingSet2_0" Index="0"/>
  <ApplicationSet Name="WebBrowsingSet2_1" Index="1"/>
</NodeContainer>
. . .
  
```

Snippet 6: Sector labeling based on latency attributes and Node types.

A stand-alone sector labeling tool assigns `Sector` labels where needed, and outputs the sector graph in METIS format. The METIS graph partitioning tool is used to assign sectors to MPI ranks based on its partitioning logic. An XSLT transform then applies the Sector-to-rank assignments reported by METIS to the original XML, by adding the `SystemId` attribute to the `<NodeContainers>` and/or `<RefNode>`s, as shown in Snippet 7. The example in Snippet 6 and Snippet 7 illustrate the possibility of more than one sector being processed on the same rank.

```

. . .
<NodeContainer Name="csma_1_nodes" SystemId="0"SystemId="1"SystemId="2"/>
    <RefNode Name="ALL_NODES" Index="1" SystemId="2"/>
    <ApplicationSet Name="WebBrowsingSet2_0" Index="0"/>
    <ApplicationSet Name="WebBrowsingSet2_1" Index="1"/>
</NodeContainer>
. . .

```

Snippet 7: SystemId attribute for rank assignment based on METIS partitioning.

This three-step process, Sector labeling, partitioning, then application of `SystemIds` to `Sectors`, facilitates reuse of the Sector labeling, which need be performed only once for a given model. The Sector-labeled XML description can be reused for different partitioning strategies, for different numbers of MPI ranks, or different partitioning functions.

6 Ongoing and Future XNDL Development

This version of the XNDL for ns-3 model is the initial version currently being used for specifying ns-3 input specifications with XML. This model will be expanded to support the ongoing studies of large-scale network simulations.



Some of the next steps for the development of the XNDL framework include:

- Condensing parameterization on elements to a more general representation enabling reuse within an XML input specification while minimizing repetition.

- Continue expanding the ns-3 specific grammar to incorporate more ns-3 model types, particularly wireless channels.
- Create the connecting tools in the XNDL framework to enable automation of simulation pre-processing – such as the METIS partitioning.
- Create a higher level network grammar to allow construction of generic models that can then be mapped to the XNDL ns-3 grammar or a different simulation implementation (refer back to Figure 1).

As XNDL evolves, so will this document to reflect the growing capabilities of the entire framework. A full reference of the XSD, including diagrams and source, is contained in Appendix A.

7 Bibliography

- [1] P. D. Barnes, B. B. Abelev, L. E. Banks, J. M. Brase, D. R. Jefferson, S. Nikolaev, S. G. Smith and R. A. Soltz, "Integrating ns-3 Model Construction, Description, Preprocessing, Execution, and Visualization," 2013.
- [2] ns-3 Network Simulator Tutorial; Release ns-3.15, ns-3 Project, 2012.
- [3] ns-3 Network Simulator Manual, Release-3.15, ns-3 Project, 2012.
- [4] ns-3 Model Library; Release ns-3.15, ns-3 Project, 2012.

Appendix A. XNDL for ns-3 v0.9 – XSD Reference

Schema NetSim.xsd

schema location: [NetSim.xsd](#)
 attributeFormDefault: **unqualified**
 elementFormDefault: **qualified**

Elements Complex types

NetSim	AddressRangeType
	ApplicationRefType
	ApplicationSetType
	AssignAppSetType
	NodeContainerType
	NodeRefType
	PartitionType
	RefNodeInterfaceDefinitionType
	SimType
	SubnetType

schema location: [NetSimShared.xsd](#)
 attributeFormDefault: **unqualified**
 elementFormDefault: **qualified**

Complex types	Simple types
AbstractApplicationType	HostTypeEnum
PromiscuousModeType	IPv4AddressType
	PercentType
	SubnetTypeEnum

schema location: [BulkSendApp.xsd](#)
 attributeFormDefault: **unqualified**
 elementFormDefault: **qualified**

Complex types

BulkSendAppType

schema location: [OnOffApp.xsd](#)
 attributeFormDefault: **unqualified**
 elementFormDefault: **qualified**

Complex types

[OnOffAppType](#)

schema location:

[PacketSinkApp.xsd](#)

attributeFormDefault:

unqualified

elementFormDefault:

qualified

Complex types

[PacketSinkAppType](#)

schema location:

[UdpClientApp.xsd](#)

attributeFormDefault:

unqualified

elementFormDefault:

qualified

Complex types

[UdpClientAppType](#)

schema location:

[UdpServerApp.xsd](#)

attributeFormDefault:

unqualified

elementFormDefault:

qualified

Complex types

[UdpServerAppType](#)

schema location:

[UdpEchoClientApp.xsd](#)

attributeFormDefault:

unqualified

elementFormDefault:

qualified

Complex types

[UdpEchoClientAppType](#)

schema location:

[UdpEchoServerApp.xsd](#)

attributeFormDefault:

unqualified

elementFormDefault:

qualified

Complex types

[UdpEchoServerAppType](#)

schema location:

[UdpTraceClientApp.xsd](#)

attributeFormDefault: **unqualified**
elementFormDefault: **qualified**

Complex types
[**UdpTraceClientAppType**](#)

schema location: [**V4PingApp.xsd**](#)
attributeFormDefault: **unqualified**
elementFormDefault: **qualified**

Complex types
[**V4PingType**](#)

schema location: [**GenericApp.xsd**](#)
attributeFormDefault: **unqualified**
elementFormDefault: **qualified**

Complex types
[**AttributeValuePairType**](#)
[**GenericAppType**](#)

schema location: [**SimpleWebClientApp.xsd**](#)
attributeFormDefault: **unqualified**
elementFormDefault: **qualified**

Complex types
[**SimpleWebClientAppType**](#)

schema location: [**SimpleWebServerApp.xsd**](#)
attributeFormDefault: **unqualified**
elementFormDefault: **qualified**

Complex types
[**SimpleWebServerAppType**](#)

element NetSim

diagram	<pre> classDiagram class NetSim { attributes Name P2pEnablePcapAll P2pEnableAsciiTraceAll CsmaEnablePcapAll CsmaEnableAsciiTraceAll SchemaVersion associations "0..∞" *-- "Sim" "0..∞" *-- "MPIEnabled" "0..∞" *-- "Partition" "1..∞" *-- "NodeContainer" "0..∞" *-- "Subnet" "0..∞" *-- "Application" "0..∞" *-- "ApplicationSet" } </pre>																																										
properties	content complex																																										
children	Sim MPIEnabled Partition NodeContainer Subnet Application ApplicationSet																																										
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>P2pEnablePcapAll</td> <td>xs:boolean</td> <td></td> <td>false</td> <td></td> <td></td> </tr> <tr> <td>P2pEnableAsciiTraceAll</td> <td>xs:boolean</td> <td></td> <td>false</td> <td></td> <td></td> </tr> <tr> <td>CsmaEnablePcapAll</td> <td>xs:boolean</td> <td></td> <td>false</td> <td></td> <td></td> </tr> <tr> <td>CsmaEnableAsciiTraceAll</td> <td>xs:boolean</td> <td></td> <td>false</td> <td></td> <td></td> </tr> <tr> <td>SchemaVersion</td> <td>xs:decimal</td> <td>required</td> <td>1.0</td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	xs:string	required				P2pEnablePcapAll	xs:boolean		false			P2pEnableAsciiTraceAll	xs:boolean		false			CsmaEnablePcapAll	xs:boolean		false			CsmaEnableAsciiTraceAll	xs:boolean		false			SchemaVersion	xs:decimal	required	1.0		
Name	Type	Use	Default	Fixed	Annotation																																						
Name	xs:string	required																																									
P2pEnablePcapAll	xs:boolean		false																																								
P2pEnableAsciiTraceAll	xs:boolean		false																																								
CsmaEnablePcapAll	xs:boolean		false																																								
CsmaEnableAsciiTraceAll	xs:boolean		false																																								
SchemaVersion	xs:decimal	required	1.0																																								
source	<pre> <xs:element name="NetSim"> <xs:complexType> <xs:sequence> <xs:element name="Sim" type="SimType" minOccurs="0"/> <xs:element name="MPIEnabled" type="xs:long" minOccurs="0"/> <xs:element name="Partition" type="PartitionType" minOccurs="0" maxOccurs="unbounded"/> </pre>																																										

	<pre> <!-- A NodeContainer element specifies a group nodes. --> <xss:element name="NodeContainer" type="NodeContainerType" maxOccurs="unbounded"/> <!-- A Subnet element defines the Node/Interface pairings and throughput characteristics for a subnet. --> <!-- Each subnet is associated, via the NodeContainerName attribute, with a single NodeContainer. --> <!-- The set of all Subnet elements define the topology of the network. --> <xss:element name="Subnet" type="SubnetType" minOccurs="0" maxOccurs="unbounded"/> <!-- The specification of an application --> <xss:element name="Application" type="AbstractApplicationType" minOccurs="0" maxOccurs="unbounded"/> <!-- A set of applications that can be installed on one of more nodes --> <xss:element name="ApplicationSet" type="ApplicationSetType" minOccurs="0" maxOccurs="unbounded"/> </xss:sequence> <xss:attribute name="Name" type="xs:string" use="required"/> <xss:attribute name="P2pEnablePcapAll" type="xs:boolean" default="false"/> <xss:attribute name="P2pEnableAsciiTraceAll" type="xs:boolean" default="false"/> <xss:attribute name="CsmaEnablePcapAll" type="xs:boolean" default="false"/> <xss:attribute name="CsmaEnableAsciiTraceAll" type="xs:boolean" default="false"/> <xss:attribute name="SchemaVersion" type="xs:decimal" use="required" fixed="1.0"/> </xss:complexType> </xss:element> </pre>
--	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

attribute NetSim/@Name

type	xs:string
properties	use required
source	<xss:attribute name="Name" type="xs:string" use="required"/>

attribute NetSim/@P2pEnablePcapAll

type	xs:boolean
properties	default false
source	<xss:attribute name="P2pEnablePcapAll" type="xs:boolean" default="false"/>

attribute NetSim/@P2pEnableAsciiTraceAll

type	xs:boolean
properties	default false
source	<xss:attribute name="P2pEnableAsciiTraceAll" type="xs:boolean" default="false"/>

attribute NetSim/@CsmaEnablePcapAll

type	xs:boolean
------	-------------------

properties	default false
source	<xs:attribute name="CsmaEnablePcapAll" type="xs:boolean" default="false"/>

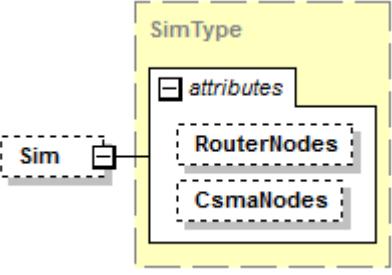
attribute **NetSim/@CsmaEnableAsciiTraceAll**

type	xs:boolean
properties	default false
source	<xs:attribute name="CsmaEnableAsciiTraceAll" type="xs:boolean" default="false"/>

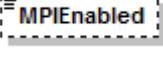
attribute **NetSim/@SchemaVersion**

type	xs:decimal
properties	use required fixed 1.0
source	<xs:attribute name="SchemaVersion" type="xs:decimal" use="required" fixed="1.0"/>

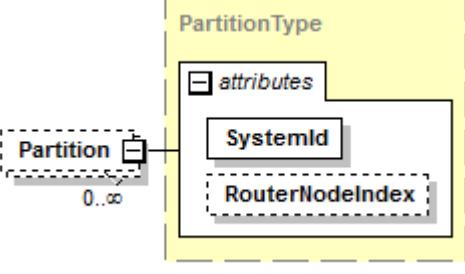
element **NetSim/Sim**

diagram	
type	SimType
properties	minOcc 0 maxOcc 1 content complex
attributes	Name Type Use Default Fixed Annotation <u>RouterNodes</u> xs:string <u>CsmaNodes</u> xs:string
source	<xs:element name="Sim" type="SimType" minOccurs="0"/>

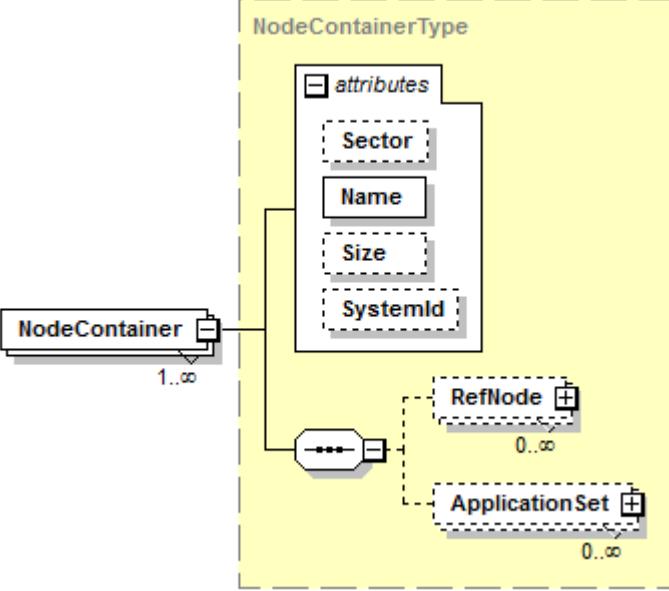
element **NetSim/MPIEnabled**

diagram	
type	xs:long
properties	minOcc 0 maxOcc 1 content simple
source	<xs:element name="MPIEnabled" type="xs:long" minOccurs="0"/>

element NetSim/Partition

diagram	
type	PartitionType
properties	minOcc 0 maxOcc unbounded content complex
attributes	Name SystemId Type xs:string Use required Default Fixed Annotation Name RouterNodeIndex Type xs:string
source	<code><xs:element name="Partition" type="PartitionType" minOccurs="0" maxOccurs="unbounded"/></code>

element NetSim/NodeContainer

diagram	
type	NodeContainerType
properties	minOcc 1 maxOcc unbounded content complex
children	RefNode ApplicationSet
attributes	Name Sector Type xs:long Use required Default Fixed Annotation Name Name Type xs:string

	<u>Size</u> xs:long 0 <u>SystemId</u> xs:long 0
source	<xs:element name="NodeContainer" type="NodeContainerType" maxOccurs="unbounded"/>

element NetSim/Subnet

diagram	<pre> classDiagram class SubnetType { Name Type NodeContainer DataRate Delay Cidr } class Subnet { <<Description>> RefNode AddressRange PromiscuousMode } SubnetType "0..∞" -- "1..∞" Subnet Subnet "1..∞" -- "1..∞" Description Subnet "1..∞" -- "0..254" RefNode Subnet "1..∞" -- "0..254" AddressRange Subnet "1..∞" -- "0..254" PromiscuousMode </pre>																																										
type	<u>SubnetType</u>																																										
properties	minOcc 0 maxOcc unbounded content complex																																										
children	<u>Description</u> <u>RefNode</u> <u>AddressRange</u> <u>PromiscuousMode</u>																																										
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td><u>Name</u></td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td><u>Type</u></td> <td><u>SubnetTypeEnum</u></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td><u>NodeContainer</u></td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td><u>DataRate</u></td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td><u>Delay</u></td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td><u>Cidr</u></td> <td>xs:string</td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	<u>Name</u>	xs:string	required				<u>Type</u>	<u>SubnetTypeEnum</u>					<u>NodeContainer</u>	xs:string	required				<u>DataRate</u>	xs:string	required				<u>Delay</u>	xs:string	required				<u>Cidr</u>	xs:string				
Name	Type	Use	Default	Fixed	Annotation																																						
<u>Name</u>	xs:string	required																																									
<u>Type</u>	<u>SubnetTypeEnum</u>																																										
<u>NodeContainer</u>	xs:string	required																																									
<u>DataRate</u>	xs:string	required																																									
<u>Delay</u>	xs:string	required																																									
<u>Cidr</u>	xs:string																																										
source	<xs:element name="Subnet" type="SubnetType" minOccurs="0" maxOccurs="unbounded"/>																																										

element NetSim/Application

diagram	<pre> classDiagram class AbstractApplicationType { <<AbstractApplicationType>> attributes +Name +Start +Stop } class Application { <<Application>> } Application "0..∞" -- "1" AbstractApplicationType </pre>																								
type	AbstractApplicationType																								
properties	minOcc 0 maxOcc unbounded content complex																								
attributes	<table> <tr> <td>Name</td> <td>Type xs:string</td> <td>Use required</td> <td>Default</td> <td>Fixed</td> <td>Annotation</td> </tr> <tr> <td>Name</td> <td>xs:string</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Start</td> <td>xs:double</td> <td>optional</td> <td>1.0</td> <td></td> <td></td> </tr> <tr> <td>Stop</td> <td>xs:double</td> <td>optional</td> <td>10.0</td> <td></td> <td></td> </tr> </table>	Name	Type xs:string	Use required	Default	Fixed	Annotation	Name	xs:string					Start	xs:double	optional	1.0			Stop	xs:double	optional	10.0		
Name	Type xs:string	Use required	Default	Fixed	Annotation																				
Name	xs:string																								
Start	xs:double	optional	1.0																						
Stop	xs:double	optional	10.0																						
source	<pre><xs:element name="Application" type="AbstractApplicationType" minOccurs="0" maxOccurs="unbounded"/></pre>																								

element NetSim/ApplicationSet

diagram	<pre> classDiagram class ApplicationSetType { <<ApplicationSetType>> attributes +Name +Percent +Function } class ApplicationSet { <<ApplicationSet>> } ApplicationSet "0..∞" -- "1" ApplicationSetType ApplicationSetType "*" -- "0..∞" Application </pre>																								
type	ApplicationSetType																								
properties	minOcc 0 maxOcc unbounded content complex																								
children	Application																								
attributes	<table> <tr> <td>Name</td> <td>Type xs:string</td> <td>Use required</td> <td>Default</td> <td>Fixed</td> <td>Annotation</td> </tr> <tr> <td>Name</td> <td>xs:string</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Percent</td> <td>PercentType</td> <td></td> <td>0</td> <td></td> <td></td> </tr> <tr> <td>Function</td> <td>xs:string</td> <td></td> <td></td> <td></td> <td></td> </tr> </table>	Name	Type xs:string	Use required	Default	Fixed	Annotation	Name	xs:string					Percent	PercentType		0			Function	xs:string				
Name	Type xs:string	Use required	Default	Fixed	Annotation																				
Name	xs:string																								
Percent	PercentType		0																						
Function	xs:string																								
source	<pre><xs:element name="ApplicationSet" type="ApplicationSetType" minOccurs="0" maxOccurs="unbounded"/></pre>																								

complexType **AddressRangeType**

diagram	A UML class diagram showing a class named 'AddressRangeType'. It has three attributes: 'Network' (solid box), 'Mask' (solid box), and 'Base' (dashed box). A legend indicates that solid boxes represent required attributes and dashed boxes represent optional attributes.																								
used by	element SubnetType/AddressRange																								
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Network</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Mask</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Base</td> <td>xs:string</td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Network	xs:string	required				Mask	xs:string	required				Base	xs:string				
Name	Type	Use	Default	Fixed	Annotation																				
Network	xs:string	required																							
Mask	xs:string	required																							
Base	xs:string																								
source	<pre><xs:complexType name="AddressRangeType"> <xs:attribute name="Network" type="xs:string" use="required"/> <xs:attribute name="Mask" type="xs:string" use="required"/> <xs:attribute name="Base" type="xs:string"/> </xs:complexType></pre>																								

attribute **AddressRangeType/@Network**

type	xs:string
properties	use required
source	<pre><xs:attribute name="Network" type="xs:string" use="required"/></pre>

attribute **AddressRangeType/@Mask**

type	xs:string
properties	use required
source	<pre><xs:attribute name="Mask" type="xs:string" use="required"/></pre>

attribute **AddressRangeType/@Base**

type	xs:string
source	<pre><xs:attribute name="Base" type="xs:string"/></pre>

complexType **ApplicationRefType**

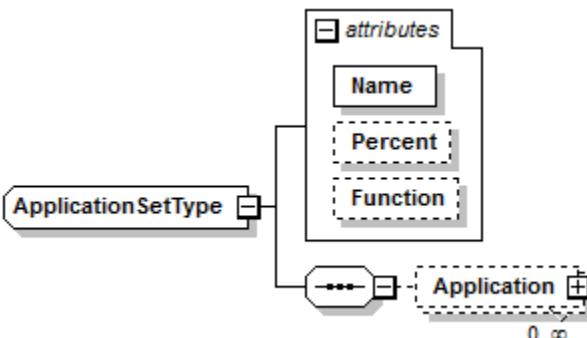
diagram	A UML class diagram showing a class named 'ApplicationRefType'. It has one attribute: 'Name' (solid box).
---------	-----------------------------------------------------------------------------------------------------------

used by	element ApplicationSetType/Application					
attributes	Name Name	Type xs:string	Use required	Default	Fixed	Annotation
source	<xs:complexType name="ApplicationRefType"> <xs:attribute name="Name" type="xs:string" use="required"/> </xs:complexType>					

attribute **ApplicationRefType/@Name**

type	xs:string
properties	use required
source	<xs:attribute name="Name" type="xs:string" use="required"/>

complexType **ApplicationSetType**

diagram	
children	Application
used by	element NetSim/ApplicationSet
attributes	Name Name Type xs:string Use required Default Percent PercentType 0 Function xs:string
source	<xs:complexType name="ApplicationSetType"> <xs:sequence> <xs:element name="Application" type="ApplicationRefType" minOccurs="0" maxOccurs="unbounded"/> </xs:sequence> <xs:attribute name="Name" type="xs:string" use="required"/> <xs:attribute name="Percent" type="PercentType" default="0"/> <xs:attribute name="Function" type="xs:string"/> <!-- Install this app set to given Percent of all nodes in the simulation. --> <!-- Install this app set to all nodes determined by supplied function name. --> </xs:complexType>

attribute **ApplicationSetType/@Name**

type	xs:string
------	------------------

properties	use required
source	<xs:attribute name="Name" type="xs:string" use="required"/>

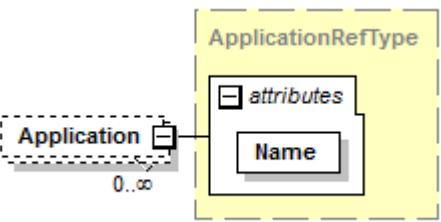
attribute **ApplicationSetType/@Percent**

type	PercentType
properties	default 0
facets	Kind Value Annotation minInclusive 0 maxInclusive 100
source	<xs:attribute name="Percent" type="PercentType" default="0"/>

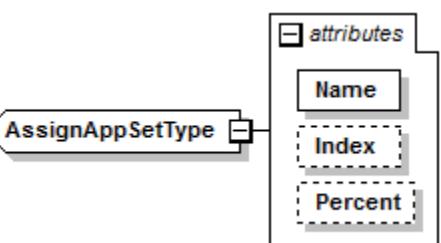
attribute **ApplicationSetType/@Function**

type	xs:string
source	<xs:attribute name="Function" type="xs:string"/>

element **ApplicationSetType/Application**

diagram	
type	ApplicationRefType
properties	minOcc 0 maxOcc unbounded content complex
attributes	Name Type Use Default Annotation Name xs:string required Fixed
source	<xs:element name="Application" type="ApplicationRefType" minOccurs="0" maxOccurs="unbounded"/>

complexType **AssignAppSetType**

diagram	
used by	element NodeContainerType/ApplicationSet

attributes	Name <u>Name</u>	Type xs:string	Use required	Default	Fixed	Annotation
source	<pre><xs:complexType name="AssignAppSetType"> <xs:attribute name="Name" type="xs:string" use="required"/> <xs:attribute name="Index" type="xs:long"/> <xs:attribute name="Percent" type="PercentType" default="0"/> <!-- the name of the ApplicationSet . --> <!-- container index that identifies the node the application set should be installed on --> <!-- appset will be installed on the given 'Percent' of randomly selected node --> </xs:complexType></pre>					

attribute AssignAppSetType/@Name

type	xs:string
properties	use required
source	<pre><xs:attribute name="Name" type="xs:string" use="required"/></pre>

attribute AssignAppSetType/@Index

type	xs:long
source	<pre><xs:attribute name="Index" type="xs:long"/></pre>

attribute AssignAppSetType/@Percent

type	PercentType
properties	default 0
facets	Kind Value Annotation minInclusive 0 maxInclusive 100
source	<pre><xs:attribute name="Percent" type="PercentType" default="0"/></pre>

complexType **NodeContainerType**

diagram																															
children	RefNode ApplicationSet																														
used by	element NetSim/NodeContainer																														
attributes	<table> <tr> <td>Name</td><td>Type</td><td>Use</td><td>Default</td><td>Fixed</td><td>Annotation</td></tr> <tr> <td>Sector</td><td>xs:long</td><td></td><td>0</td><td></td><td></td></tr> <tr> <td>Name</td><td>xs:string</td><td>required</td><td></td><td></td><td></td></tr> <tr> <td>Size</td><td>xs:long</td><td></td><td>0</td><td></td><td></td></tr> <tr> <td>SystemId</td><td>xs:long</td><td></td><td>0</td><td></td><td></td></tr> </table>	Name	Type	Use	Default	Fixed	Annotation	Sector	xs:long		0			Name	xs:string	required				Size	xs:long		0			SystemId	xs:long		0		
Name	Type	Use	Default	Fixed	Annotation																										
Sector	xs:long		0																												
Name	xs:string	required																													
Size	xs:long		0																												
SystemId	xs:long		0																												
source	<pre> <xs:complexType name="NodeContainerType"> <xs:sequence> <!-- the RefNode element adds a node previously defined in another container to this container. --> <xs:element name="RefNode" type="NodeRefType" minOccurs="0" maxOccurs="unbounded"/> <!-- the assignAppSet adds an applicationSet to either a specific node or a percentage of randomly selected nodes in this container --> <xs:element name="ApplicationSet" type="AssignAppSetType" minOccurs="0" maxOccurs="unbounded"/> </xs:sequence> <xs:attribute name="Sector" type="xs:long" default="0"/> <xs:attribute name="Name" type="xs:string" use="required"/> <xs:attribute name="Size" type="xs:long" default="0"/> <xs:attribute name="SystemId" type="xs:long" default="0"/> <!-- number of nodes to create --> <!-- to enable MPI runs --> </xs:complexType></pre>																														

attribute **NodeContainerType/@Sector**

type	xs:long
properties	default 0
source	<xs:attribute name="Sector" type="xs:long" default="0"/>

attribute **NodeContainerType/@Name**

type	xs:string
properties	use required
source	<xs:attribute name="Name" type="xs:string" use="required"/>

attribute **NodeContainerType/@Size**

type	xs:long
properties	default 0
source	<xs:attribute name="Size" type="xs:long" default="0"/>

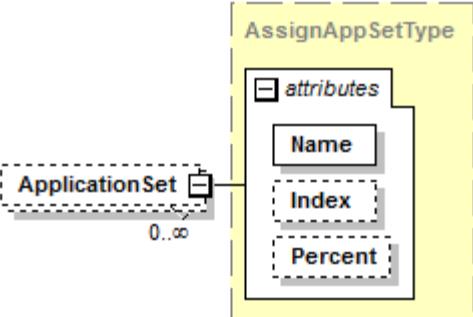
attribute **NodeContainerType/@SystemId**

type	xs:long
properties	default 0
source	<xs:attribute name="SystemId" type="xs:long" default="0"/>

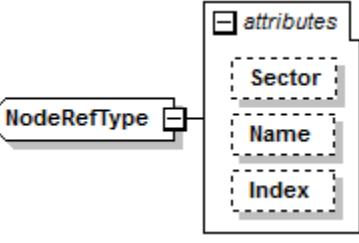
element **NodeContainerType/RefNode**

diagram	<pre> classDiagram class RefNode { <<0..>> } class NodeRefType { <<attributes>> class Sector class Name class Index } RefNode --> NodeRefType </pre>																								
type	NodeRefType																								
properties	minOcc 0 maxOcc unbounded content complex																								
attributes	<table> <tr> <td>Name</td> <td>Type</td> <td>Use</td> <td>Default</td> <td>Fixed</td> <td>Annotation</td> </tr> <tr> <td><u>Sector</u></td> <td>xs:long</td> <td></td> <td>0</td> <td></td> <td></td> </tr> <tr> <td><u>Name</u></td> <td>xs:string</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td><u>Index</u></td> <td>xs:long</td> <td></td> <td></td> <td></td> <td></td> </tr> </table>	Name	Type	Use	Default	Fixed	Annotation	<u>Sector</u>	xs:long		0			<u>Name</u>	xs:string					<u>Index</u>	xs:long				
Name	Type	Use	Default	Fixed	Annotation																				
<u>Sector</u>	xs:long		0																						
<u>Name</u>	xs:string																								
<u>Index</u>	xs:long																								
source	<xs:element name="RefNode" type="NodeRefType" minOccurs="0" maxOccurs="unbounded"/>																								

element **NodeContainerType/ApplicationSet**

diagram																									
type	AssignAppSetType																								
properties	minOcc 0 maxOcc unbounded content complex																								
attributes	<table> <tr> <td>Name</td> <td>Type xs:string</td> <td>Use required</td> <td>Default</td> <td>Fixed</td> <td>Annotation</td> </tr> <tr> <td>Name</td> <td>xs:string</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Index</td> <td>xs:long</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Percent</td> <td>PercentType</td> <td></td> <td>0</td> <td></td> <td></td> </tr> </table>	Name	Type xs:string	Use required	Default	Fixed	Annotation	Name	xs:string					Index	xs:long					Percent	PercentType		0		
Name	Type xs:string	Use required	Default	Fixed	Annotation																				
Name	xs:string																								
Index	xs:long																								
Percent	PercentType		0																						
source	<pre><xs:element name="ApplicationSet" type="AssignAppSetType" minOccurs="0" maxOccurs="unbounded"/></pre>																								

complexType **NodeRefType**

diagram																									
used by	element NodeContainerType/RefNode																								
attributes	<table> <tr> <td>Name</td> <td>Type xs:long</td> <td>Use</td> <td>Default</td> <td>Fixed</td> <td>Annotation</td> </tr> <tr> <td>Sector</td> <td>xs:long</td> <td></td> <td>0</td> <td></td> <td></td> </tr> <tr> <td>Name</td> <td>xs:string</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Index</td> <td>xs:long</td> <td></td> <td></td> <td></td> <td></td> </tr> </table>	Name	Type xs:long	Use	Default	Fixed	Annotation	Sector	xs:long		0			Name	xs:string					Index	xs:long				
Name	Type xs:long	Use	Default	Fixed	Annotation																				
Sector	xs:long		0																						
Name	xs:string																								
Index	xs:long																								
source	<pre><xs:complexType name="NodeRefType"> <xs:attribute name="Sector" type="xs:long" default="0"/> <xs:attribute name="Name" type="xs:string"/> <xs:attribute name="Index" type="xs:long"/> <!-- the name of the container containing the node. --> <!-- container index that identifies the node --> </xs:complexType></pre>																								

attribute **NodeRefType/@Sector**

type	xs:long
------	-------------------------

properties	default 0
source	<xs:attribute name="Sector" type="xs:long" default="0"/>

attribute NodeRefType/@Name

type	xs:string
source	<xs:attribute name="Name" type="xs:string"/>

attribute NodeRefType/@Index

type	xs:long
source	<xs:attribute name="Index" type="xs:long"/>

complexType PartitionType

diagram	<pre> classDiagram class PartitionType { attribute SystemId : xs:string attribute RouterNodeIndex : xs:string } </pre>																		
used by	element NetSim/Partition																		
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>SystemId</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>RouterNodeIndex</td> <td>xs:string</td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	SystemId	xs:string	required				RouterNodeIndex	xs:string				
Name	Type	Use	Default	Fixed	Annotation														
SystemId	xs:string	required																	
RouterNodeIndex	xs:string																		
source	<xs:complexType name="PartitionType"> <xs:attribute name="SystemId" type="xs:string" use="required"/> <xs:attribute name="RouterNodeIndex" type="xs:string"/> </xs:complexType>																		

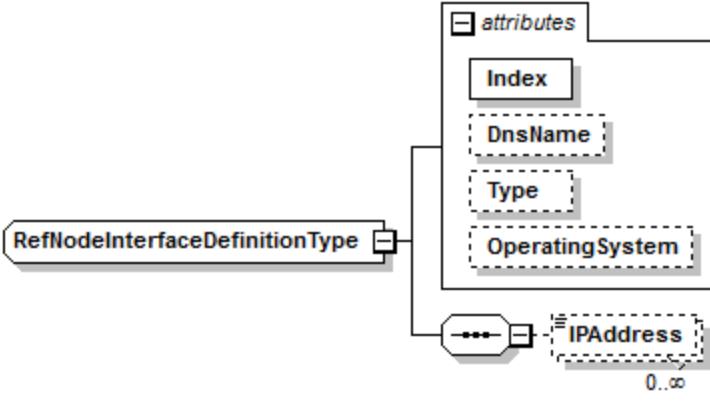
attribute PartitionType/@SystemId

type	xs:string
properties	use required
source	<xs:attribute name="SystemId" type="xs:string" use="required"/>

attribute PartitionType/@RouterNodeIndex

type	xs:string
source	<xs:attribute name="RouterNodeIndex" type="xs:string"/>

complexType RefNodeInterfaceDefinitionType

diagram																															
children	IPAddress																														
used by	element SubnetType/RefNode																														
attributes	<table> <tr> <td>Name</td> <td>Type</td> <td>Use</td> <td>Default</td> <td>Fixed</td> <td>Annotation</td> </tr> <tr> <td>Index</td> <td>xs:long</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>DnsName</td> <td>xs:string</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Type</td> <td>HostTypeEnum</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>OperatingSystem</td> <td>xs:string</td> <td></td> <td></td> <td></td> <td></td> </tr> </table>	Name	Type	Use	Default	Fixed	Annotation	Index	xs:long	required				DnsName	xs:string					Type	HostTypeEnum					OperatingSystem	xs:string				
Name	Type	Use	Default	Fixed	Annotation																										
Index	xs:long	required																													
DnsName	xs:string																														
Type	HostTypeEnum																														
OperatingSystem	xs:string																														
source	<pre><xs:complexType name="RefNodeInterfaceDefinitionType"> <xs:sequence> <xs:element name="IPAddress" type="IPv4AddressType" minOccurs="0" maxOccurs="unbounded"/> </xs:sequence> <xs:attribute name="Index" type="xs:long" use="required"/> <xs:attribute name="DnsName" type="xs:string"/> <xs:attribute name="Type" type="HostTypeEnum"/> <xs:attribute name="OperatingSystem" type="xs:string"/> <!-- container index that identifies the node the above IPAddresses should be assigned to --> </xs:complexType></pre>																														

attribute RefNodeInterfaceDefinitionType/@Index

type	xs:long
properties	use required
source	<xs:attribute name="Index" type="xs:long" use="required"/>

attribute RefNodeInterfaceDefinitionType/@DnsName

type	xs:string
source	<xs:attribute name="DnsName" type="xs:string"/>

attribute RefNodeInterfaceDefinitionType/@Type

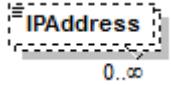
type	HostTypeEnum
------	------------------------------

facets	Kind enumeration Value SIMPLE enumeration SWITCH enumeration ROUTER
source	<xs:attribute name="Type" type="HostTypeEnum"/>

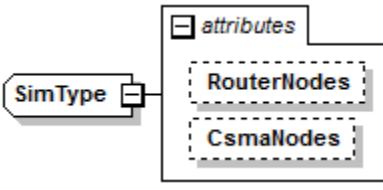
attribute RefNodeInterfaceDefinitionType/@OperatingSystem

type	xs:string
source	<xs:attribute name="OperatingSystem" type="xs:string"/>

element RefNodeInterfaceDefinitionType/IPAddress

diagram	
type	IPv4AddressType
properties	minOcc 0 maxOcc unbounded content simple
facets	Kind Value pattern ((25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[1-9][0-9]) (0-9){3})((25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[1-9][0-9]) (0-9){3}) Annotation
source	<xs:element name="IPAddress" type="IPv4AddressType" minOccurs="0" maxOccurs="unbounded"/>

complexType SimType

diagram	
used by	element NetSim/Sim
attributes	Name Type Use Default Fixed Annotation <u>RouterNodes</u> xs:string <u>CsmaNodes</u> xs:string
source	<xs:complexType name="SimType"> <xs:attribute name="RouterNodes" type="xs:string"/> <xs:attribute name="CsmaNodes" type="xs:string"/> </xs:complexType>

attribute SimType/@RouterNodes

type	xs:string
source	<xs:attribute name="RouterNodes" type="xs:string"/>

attribute **SimType/@CsmaNodes**

type	<code>xs:string</code>
source	<code><xs:attribute name="CsmaNodes" type="xs:string"/></code>

complexType **SubnetType**

diagram	<pre> classDiagram class SubnetType { attribute Name attribute Type attribute NodeContainer attribute DataRate attribute Delay attribute Cidr sequenceRef RefNode* sequenceRef AddressRange attribute PromiscuousMode } class RefNode class AddressRange </pre>																																										
children	Description RefNode AddressRange PromiscuousMode																																										
used by	element NetSim/Subnet																																										
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td><code>xs:string</code></td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Type</td> <td>SubnetTypeEnum</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>NodeContainer</td> <td><code>xs:string</code></td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>DataRate</td> <td><code>xs:string</code></td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Delay</td> <td><code>xs:string</code></td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Cidr</td> <td><code>xs:string</code></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	<code>xs:string</code>	required				Type	SubnetTypeEnum					NodeContainer	<code>xs:string</code>	required				DataRate	<code>xs:string</code>	required				Delay	<code>xs:string</code>	required				Cidr	<code>xs:string</code>				
Name	Type	Use	Default	Fixed	Annotation																																						
Name	<code>xs:string</code>	required																																									
Type	SubnetTypeEnum																																										
NodeContainer	<code>xs:string</code>	required																																									
DataRate	<code>xs:string</code>	required																																									
Delay	<code>xs:string</code>	required																																									
Cidr	<code>xs:string</code>																																										
source	<pre> <xs:complexType name="SubnetType"> <xs:sequence> <xs:element name="Description" type="xs:string" minOccurs="0"/> <xs:choice> <!-- Use sequence of RefNodes when explicitly assigning addresses to Nodes in the subnet --> <xs:element name="RefNode" type="RefNodeInterfaceDefinitionType" maxOccurs="unbounded"/> <!-- Use AddressRange when defining a sequentially generated set of addresses to --> </xs:choice> </xs:sequence> </xs:complexType> </pre>																																										

	<pre> assign to nodes in the subnet --> <xs:element name="AddressRange" type="AddressRangeType"/> </xs:choice> <!-- This is an optional list of indexes of interfaces within the defined network range that should be put into promiscuous mode --> <xs:element name="PromiscuousMode" type="PromiscuousModeType" minOccurs="0" maxOccurs="254"/> </xs:sequence> <xs:attribute name="Name" type="xs:string" use="required"/> <xs:attribute name="Type" type="SubnetTypeEnum"/> <xs:attribute name="NodeContainer" type="xs:string" use="required"/> <xs:attribute name="DataRate" type="xs:string" use="required"/> <xs:attribute name="Delay" type="xs:string" use="required"/> <xs:attribute name="Cidr" type="xs:string"/> </xs:complexType> </pre>
--	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

attribute **SubnetType/@Name**

type	xs:string
properties	use required
source	<xs:attribute name="Name" type="xs:string" use="required"/>

attribute **SubnetType/@Type**

type	SubnetTypeEnum									
facets	<table> <thead> <tr> <th>Kind</th> <th>Value</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>enumeration</td> <td>CSMA</td> <td></td> </tr> <tr> <td>enumeration</td> <td>P2P</td> <td></td> </tr> </tbody> </table>	Kind	Value	Annotation	enumeration	CSMA		enumeration	P2P	
Kind	Value	Annotation								
enumeration	CSMA									
enumeration	P2P									
source	<xs:attribute name="Type" type="SubnetTypeEnum"/>									

attribute **SubnetType/@NodeContainer**

type	xs:string
properties	use required
source	<xs:attribute name="NodeContainer" type="xs:string" use="required"/>

attribute **SubnetType/@DataRate**

type	xs:string
properties	use required
source	<xs:attribute name="DataRate" type="xs:string" use="required"/>

attribute **SubnetType/@Delay**

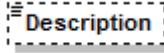
type	xs:string
properties	use required

source	<code><xs:attribute name="Delay" type="xs:string" use="required"/></code>
--------	---------------------------------------------------------------------------------

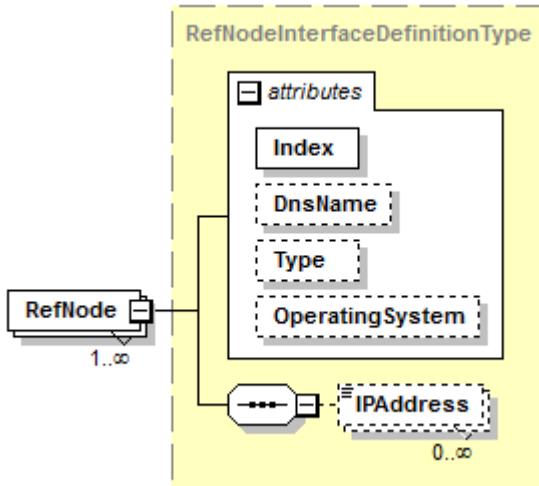
attribute **SubnetType/@Cidr**

type	<code>xs:string</code>
source	<code><xs:attribute name="Cidr" type="xs:string"/></code>

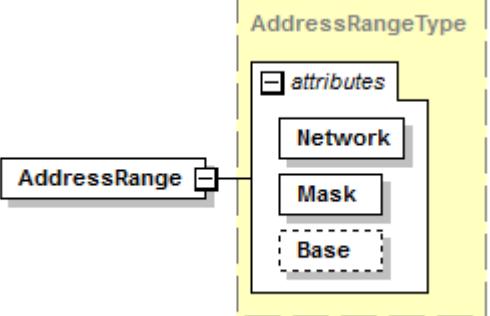
element **SubnetType/Description**

diagram	
type	<code>xs:string</code>
properties	minOcc 0 maxOcc 1 content simple
source	<code><xs:element name="Description" type="xs:string" minOccurs="0"/></code>

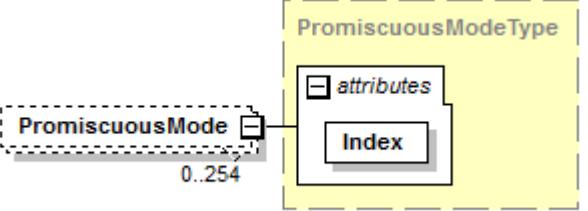
element **SubnetType/RefNode**

diagram																															
type	<code>RefNodeInterfaceDefinitionType</code>																														
properties	minOcc 1 maxOcc unbounded content complex																														
children	<code>IPAddress</code>																														
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Index</td> <td><code>xs:long</code></td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>DnsName</td> <td><code>xs:string</code></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Type</td> <td>HostTypeEnum</td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>OperatingSystem</td> <td><code>xs:string</code></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Index	<code>xs:long</code>	required				DnsName	<code>xs:string</code>					Type	HostTypeEnum					OperatingSystem	<code>xs:string</code>				
Name	Type	Use	Default	Fixed	Annotation																										
Index	<code>xs:long</code>	required																													
DnsName	<code>xs:string</code>																														
Type	HostTypeEnum																														
OperatingSystem	<code>xs:string</code>																														
source	<code><xs:element name="RefNode" type="RefNodeInterfaceDefinitionType" maxOccurs="unbounded"/></code>																														

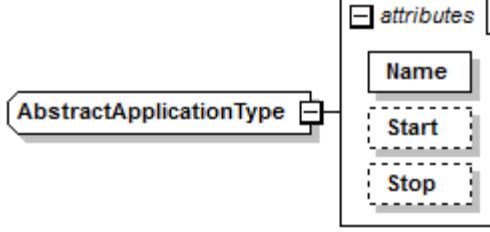
element SubnetType/AddressRange

diagram																									
type	AddressRangeType																								
properties	content complex																								
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Network</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Mask</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Base</td> <td>xs:string</td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Network	xs:string	required				Mask	xs:string	required				Base	xs:string				
Name	Type	Use	Default	Fixed	Annotation																				
Network	xs:string	required																							
Mask	xs:string	required																							
Base	xs:string																								
source	<code><xs:element name="AddressRange" type="AddressRangeType"/></code>																								

element SubnetType/PromiscuousMode

diagram													
type	PromiscuousModeType												
properties	minOcc 0 maxOcc 254 content complex												
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Index</td> <td>xs:long</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Index	xs:long	required			
Name	Type	Use	Default	Fixed	Annotation								
Index	xs:long	required											
source	<code><xs:element name="PromiscuousMode" type="PromiscuousModeType" minOccurs="0" maxOccurs="254"/></code>												

complexType **AbstractApplicationType**

diagram	
properties	abstract true
used by	element complexTypes NetSim/Application BulkSendAppType GenericAppType OnOffAppType PacketSinkAppType SimpleWebClientAppType SimpleWebServerAppType UdpClientAppType UdpEchoClientAppType UdpEchoServerAppType UdpServerAppType UdpTraceClientAppType V4PingType
attributes	Name Type xs:string Use required Default Fixed Annotation <u>Name</u> xs:string required <u>Start</u> xs:double optional 1.0 <u>Stop</u> xs:double optional 10.0
source	<pre><xs:complexType name="AbstractApplicationType" abstract="true"> <xs:attribute name="Name" type="xs:string" use="required"/> <xs:attribute name="Start" type="xs:double" use="optional" default="1.0"/> <xs:attribute name="Stop" type="xs:double" use="optional" default="10.0"/> <!-- start time in seconds --> <!-- end time in seconds --> </xs:complexType></pre>

attribute **AbstractApplicationType/@Name**

type	xs:string
properties	use required
source	<xs:attribute name="Name" type="xs:string" use="required"/>

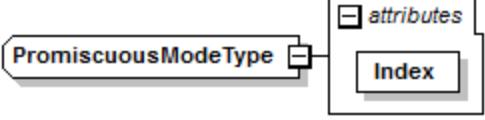
attribute **AbstractApplicationType/@Start**

type	xs:double
properties	use optional default 1.0
source	<xs:attribute name="Start" type="xs:double" use="optional" default="1.0"/>

attribute **AbstractApplicationType/@Stop**

type	xs:double
properties	use optional default 10.0
source	<xs:attribute name="Stop" type="xs:double" use="optional" default="10.0"/>

complexType PromiscuousModeType

diagram	
used by	element SubnetType/PromiscuousMode
attributes	Name Type Use Default Fixed Annotation <u>Index</u> xs:long required
source	<pre><xs:complexType name="PromiscuousModeType"> <xs:attribute name="Index" type="xs:long" use="required"/> <!-- container index that identifies the node that should be put in promiscuous mode --> </xs:complexType></pre>

attribute PromiscuousModeType/@Index

type	xs:long
properties	use required
source	<code><xs:attribute name="Index" type="xs:long" use="required"/></code>

simpleType HostTypeEnum

type	restriction of xs:string
properties	base xs:string
used by	attribute RefNodeInterfaceDefinitionType/@Type
facets	Kind Value Annotation enumeration SIMPLE enumeration SWITCH enumeration ROUTER
source	<pre><xs:simpleType name="HostTypeEnum"> <xs:restriction base="xs:string"> <xs:enumeration value="SIMPLE"/> <xs:enumeration value="SWITCH"/> <xs:enumeration value="ROUTER"/> </xs:restriction> </xs:simpleType></pre>

simpleType IPv4AddressType

type	restriction of xs:string
properties	base xs:string
used by	element RefNodeInterfaceDefinitionType/IP Address attributes PacketSinkAppType/@LocalAddress BulkSendAppType/@RemoteAddress OnOffAppType/@RemoteAddress UdpClientAppType/@RemoteAddress UdpEchoClientAppType/@RemoteAddress UdpTraceClientAppType/@RemoteAddress V4PingType/@RemoteAddress SimpleWebClientAppType/@RemoteAddress

facets	Kind pattern ((25[0-5]2[0-4][0-9]1[0-9][0-9][1-9][0-9])[0-9])\.{3}(25[0-5]2[0-4][0-9]1[0-9][0-9][1-9][0-9])[0-9])	Value Annotation
annotation	documentation	IPv4 address in the dotted-decimal notation.
source	<pre><xs:simpleType name="IPv4AddressType"> <xs:annotation> <xs:documentation> IPv4 address in the dotted-decimal notation. </xs:documentation> </xs:annotation> <xs:restriction base="xs:string"> <xs:pattern value="((25[0-5]2[0-4][0-9]1[0-9][0-9][1-9][0-9])[0-9])\.{3}(25[0-5]2[0-4][0-9]1[0-9][0-9][1-9][0-9])[0-9]"> </xs:restriction> </xs:simpleType></pre>	

simpleType PercentType

type	restriction of <code>xs:long</code>
properties	base <code>xs:long</code>
used by	attributes AssignAppSetType/@Percent ApplicationSetType/@Percent
facets	Kind Value Annotation minInclusive 0 maxInclusive 100
source	<pre><xs:simpleType name="PercentType"> <xs:restriction base="xs:long"> <xs:minInclusive value="0"/> <xs:maxInclusive value="100"/> </xs:restriction> </xs:simpleType></pre>

simpleType SubnetTypeEnum

type	restriction of <code>xs:string</code>
properties	base <code>xs:string</code>
used by	attribute SubnetType/@Type
facets	Kind Value Annotation enumeration CSMA enumeration P2P
source	<pre><xs:simpleType name="SubnetTypeEnum"> <xs:restriction base="xs:string"> <xs:enumeration value="CSMA"/> <xs:enumeration value="P2P"/> </xs:restriction> </xs:simpleType></pre>

complexType BulkSendAppType

diagram	<pre> classDiagram class BulkSendAppType class AbstractApplicationType { <<extension>> <<attributes>> Name Start Stop <</attributes>> } BulkSendAppType < --> AbstractApplicationType class BulkSendAppType { <<attributes>> Protocol SendSize MaxBytes RemoteAddress Port <</attributes>> } </pre>																																																						
type	extension of AbstractApplicationType																																																						
properties	base AbstractApplicationType																																																						
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Start</td> <td>xs:double</td> <td>optional</td> <td>1.0</td> <td></td> <td></td> </tr> <tr> <td>Stop</td> <td>xs:double</td> <td>optional</td> <td>10.0</td> <td></td> <td></td> </tr> <tr> <td>Protocol</td> <td>xs:string</td> <td></td> <td>ns3::TcpSocketFactory</td> <td></td> <td></td> </tr> <tr> <td>SendSize</td> <td>xs:string</td> <td></td> <td>512</td> <td></td> <td></td> </tr> <tr> <td>MaxBytes</td> <td>xs:string</td> <td></td> <td>0</td> <td></td> <td></td> </tr> <tr> <td>RemoteAddress</td> <td>IPv4AddressType</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Port</td> <td>xs:long</td> <td></td> <td>21</td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	xs:string	required				Start	xs:double	optional	1.0			Stop	xs:double	optional	10.0			Protocol	xs:string		ns3::TcpSocketFactory			SendSize	xs:string		512			MaxBytes	xs:string		0			RemoteAddress	IPv4AddressType	required				Port	xs:long		21		
Name	Type	Use	Default	Fixed	Annotation																																																		
Name	xs:string	required																																																					
Start	xs:double	optional	1.0																																																				
Stop	xs:double	optional	10.0																																																				
Protocol	xs:string		ns3::TcpSocketFactory																																																				
SendSize	xs:string		512																																																				
MaxBytes	xs:string		0																																																				
RemoteAddress	IPv4AddressType	required																																																					
Port	xs:long		21																																																				
source	<pre> <xs:complexType name="BulkSendAppType"> <xs:complexContent> <xs:extension base="AbstractApplicationType"> <xs:attribute name="Protocol" type="xs:string" default="ns3::TcpSocketFactory"/> <xs:attribute name="SendSize" type="xs:string" default="512"/> <xs:attribute name="MaxBytes" type="xs:string" default="0"/> <xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/> <xs:attribute name="Port" type="xs:long" default="21"/> </xs:extension> <!-- The type of Protocol to use. --> <!-- The amount of data to send each time. --> <!-- The total number of bytes to send. Once these bytes are sent, no --> <!-- packet is sent again, even in on state. The value zero means that --> <!-- there is no limit. --> <!-- Address of remote destination . --> <!-- remote Port. --> </pre>																																																						

```
</xs:complexContent>  
</xs:complexType>
```

attribute **BulkSendAppType/@Protocol**

type	xs:string
properties	default ns3::TcpSocketFactory
source	<code><xs:attribute name="Protocol" type="xs:string" default="ns3::TcpSocketFactory"/></code>

attribute **BulkSendAppType/@SendSize**

type	xs:string
properties	default 512
source	<xs:attribute name="SendSize" type="xs:string" default="512"/>

attribute **BulkSendAppType/@MaxBytes**

type	xs:string
properties	default 0
source	<xs:attribute name="MaxBytes" type="xs:string" default="0"/>

attribute **BulkSendAppType/@RemoteAddress**

type	IPv4AddressType		
properties	use required		
facets	Kind pattern	Value ((25[0-5][2[0-4][0-9]] 1[0-9][0-9])[1-9][0-9]) (0-9){3}(25[0-5][2[0-4][0-9]] 1[0-9][0-9])[1-9][0-9]) ([1-9][0-9][0-9])	Annotation
source	<xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/>		

attribute **BulkSendAppType/@Port**

type	xs:long
properties	default 21
source	<code><xs:attribute name="Port" type="xs:long" default="21"/></code>

complexType **OnOffAppType**

diagram	<pre> classDiagram class OnOffAppType class AbstractApplicationType { <<extension>> <<attributes>> Name Start Stop <</attributes>> } OnOffAppType --> AbstractApplicationType class attributes { <<Protocol>> <<DataRate>> <<PacketSize>> <<MaxBytes>> <<RemoteAddress>> <<Port>> } </pre>																																																												
type	extension of AbstractApplicationType																																																												
properties	base AbstractApplicationType																																																												
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Start</td> <td>xs:double</td> <td>optional</td> <td>1.0</td> <td></td> <td></td> </tr> <tr> <td>Stop</td> <td>xs:double</td> <td>optional</td> <td>10.0</td> <td></td> <td></td> </tr> <tr> <td>Protocol</td> <td>xs:string</td> <td></td> <td>ns3::UdpSocketFactory</td> <td></td> <td></td> </tr> <tr> <td>DataRate</td> <td>xs:string</td> <td></td> <td>500kb/s</td> <td></td> <td></td> </tr> <tr> <td>PacketSize</td> <td>xs:string</td> <td></td> <td>512</td> <td></td> <td></td> </tr> <tr> <td>MaxBytes</td> <td>xs:string</td> <td></td> <td>0</td> <td></td> <td></td> </tr> <tr> <td>RemoteAddress</td> <td>IPv4AddressType</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Port</td> <td>xs:long</td> <td></td> <td>9</td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	xs:string	required				Start	xs:double	optional	1.0			Stop	xs:double	optional	10.0			Protocol	xs:string		ns3::UdpSocketFactory			DataRate	xs:string		500kb/s			PacketSize	xs:string		512			MaxBytes	xs:string		0			RemoteAddress	IPv4AddressType	required				Port	xs:long		9		
Name	Type	Use	Default	Fixed	Annotation																																																								
Name	xs:string	required																																																											
Start	xs:double	optional	1.0																																																										
Stop	xs:double	optional	10.0																																																										
Protocol	xs:string		ns3::UdpSocketFactory																																																										
DataRate	xs:string		500kb/s																																																										
PacketSize	xs:string		512																																																										
MaxBytes	xs:string		0																																																										
RemoteAddress	IPv4AddressType	required																																																											
Port	xs:long		9																																																										
source	<pre> <xs:complexType name="OnOffAppType"> <xs:complexContent> <xs:extension base="AbstractApplicationType"> <xs:attribute name="Protocol" type="xs:string" default="ns3::UdpSocketFactory"/> <xs:attribute name="DataRate" type="xs:string" default="500kb/s"/> <xs:attribute name="PacketSize" type="xs:string" default="512"/> <xs:attribute name="MaxBytes" type="xs:string" default="0"/> <xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/> <xs:attribute name="Port" type="xs:long" default="9"/> </xs:extension> <!-- The type of Protocol to use. --> <!-- The data rate in on state. --> <!-- The size of packets sent in on state. --> </xs:complexContent> </xs:complexType> </pre>																																																												

	<pre> <!-- The total number of bytes to send. Once these bytes are sent, no --> <!-- packet is sent again, even in on state. The value zero means that --> <!-- there is no limit. --> <!-- Address of remote destination . --> <!-- remote Port. --> </xs:complexContent> </xs:complexType></pre>
--	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

attribute OnOffAppType/@Protocol

type	xs:string
properties	default ns3::UdpSocketFactory
source	<xs:attribute name="Protocol" type="xs:string" default="ns3::UdpSocketFactory"/>

attribute OnOffAppType/@DataRate

type	xs:string
properties	default 500kb/s
source	<xs:attribute name="DataRate" type="xs:string" default="500kb/s"/>

attribute OnOffAppType/@PacketSize

type	xs:string
properties	default 512
source	<xs:attribute name="PacketSize" type="xs:string" default="512"/>

attribute OnOffAppType/@MaxBytes

type	xs:string
properties	default 0
source	<xs:attribute name="MaxBytes" type="xs:string" default="0"/>

attribute OnOffAppType/@RemoteAddress

type	IPv4AddressType
properties	use required
facets	Kind Value pattern ((25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[1-9][0-9] 1[0-9])\.){3}(25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[1-9][0-9] 1[0-9]) Annotation
source	<xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/>

attribute OnOffAppType/@Port

type	xs:long
------	----------------

properties	default 9
source	<xs:attribute name="Port" type="xs:long" default="9"/>

complexType **PacketSinkAppType**

diagram	<pre> classDiagram class PacketSinkAppType { Name Start Stop Protocol LocalAddress LocalPort } class AbstractApplicationType { attributes Name Start Stop } PacketSinkAppType < -- AbstractApplicationType </pre>																																										
type	extension of AbstractApplicationType																																										
properties	base AbstractApplicationType																																										
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Start</td> <td>xs:double</td> <td>optional</td> <td>1.0</td> <td></td> <td></td> </tr> <tr> <td>Stop</td> <td>xs:double</td> <td>optional</td> <td>10.0</td> <td></td> <td></td> </tr> <tr> <td>Protocol</td> <td>xs:string</td> <td></td> <td>ns3::UdpSocketFactory</td> <td></td> <td></td> </tr> <tr> <td>LocalAddress</td> <td>IPv4AddressType</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>LocalPort</td> <td>xs:long</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	xs:string	required				Start	xs:double	optional	1.0			Stop	xs:double	optional	10.0			Protocol	xs:string		ns3::UdpSocketFactory			LocalAddress	IPv4AddressType	required				LocalPort	xs:long	required			
Name	Type	Use	Default	Fixed	Annotation																																						
Name	xs:string	required																																									
Start	xs:double	optional	1.0																																								
Stop	xs:double	optional	10.0																																								
Protocol	xs:string		ns3::UdpSocketFactory																																								
LocalAddress	IPv4AddressType	required																																									
LocalPort	xs:long	required																																									
source	<pre> <xs:complexType name="PacketSinkAppType"> <xs:complexContent> <xs:extension base="AbstractApplicationType"> <xs:attribute name="Protocol" type="xs:string" default="ns3::UdpSocketFactory"/> <xs:attribute name="LocalAddress" type="IPv4AddressType" use="required"/> <xs:attribute name="LocalPort" type="xs:long" use="required"/> </xs:extension> <!-- The type of Protocol to use. --> <!-- the local address on which to Bind the rx socket.--> <!-- the local port on which to Bind the rx socket.--> </xs:complexContent> </xs:complexType> </pre>																																										

attribute **PacketSinkAppType/@Protocol**

type	xs:string
properties	default ns3::UdpSocketFactory
source	<xs:attribute name="Protocol" type="xs:string" default="ns3::UdpSocketFactory"/>

attribute **PacketSinkAppType/@LocalAddress**

type	IPv4AddressType
properties	use required
facets	Kind Value pattern ((25[0-5] 2[0-4][0-9] 1[0-9][0-9])[0-9]\.){3}(25[0-5] 2[0-4][0-9] 1[0-9][0-9])[0-9] (1-9)[0-9]\ (0-9)) Annotation
source	<xs:attribute name="LocalAddress" type="IPv4AddressType" use="required"/>

attribute **PacketSinkAppType/@LocalPort**

type	xs:long
properties	use required
source	<xs:attribute name="LocalPort" type="xs:long" use="required"/>

complexType **UdpClientAppType**

diagram	<pre> classDiagram class UdpClientAppType { <<AbstractApplicationType extension>> <<attributes>> Name Start Stop MaxPackets Interval RemoteAddress RemotePort PacketSize } class AbstractApplicationType { <<attributes>> <<Name>> <<Start>> <<Stop>> } UdpClientAppType < -- AbstractApplicationType </pre>
type	extension of AbstractApplicationType
properties	base AbstractApplicationType

	Name	Type	Use	Default	Fixed	Annotation
attributes	<u>Name</u>	xs:string	required			
	<u>Start</u>	xs:double	optional	1.0		
	<u>Stop</u>	xs:double	optional	10.0		
	<u>MaxPackets</u>	xs:long		100		
	<u>Interval</u>	xs:double		1.0		
	<u>RemoteAddress</u>	IPv4AddressType	required			
	<u>RemotePort</u>	xs:long		0		
	<u>PacketSize</u>	xs:long		100		
source	<pre> <xs:complexType name="UdpClientAppType"> <xs:complexContent> <xs:extension base="AbstractApplicationType"> <xs:attribute name="MaxPackets" type="xs:long" default="100"/> <xs:attribute name="Interval" type="xs:double" default="1.0"/> <xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/> <xs:attribute name="RemotePort" type="xs:long" default="0"/> <xs:attribute name="PacketSize" type="xs:long" default="100"/> </xs:extension> <!-- The maximum number of packets the application will send --> <!-- The time to wait between packets --> <!-- Address of remote destination . --> <!--The destination Port of the outbound packets. --> <!-- Size of echo data in outbound packets --> </xs:complexContent> </xs:complexType> </pre>					

attribute UdpClientAppType/@MaxPackets

type	xs:long
properties	default 100
source	<xs:attribute name="MaxPackets" type="xs:long" default="100"/>

attribute UdpClientAppType/@Interval

type	xs:double
properties	default 1.0
source	<xs:attribute name="Interval" type="xs:double" default="1.0"/>

attribute UdpClientAppType/@RemoteAddress

type	IPv4AddressType		
properties	use required		
facets	Kind pattern Value ((25[0-5] 2[0-4][0-9] 1[0-9][0-9])[1-9][0-9]) (.)\.(3)(25[0-5] 2[0-4][0-9] 1[0-9][0-9]) ([1-9][0-9]) ([0-9])	Annotation	
source	<xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/>		

attribute **UdpClientAppType/@RemotePort**

type	xs:long
properties	default 0
source	<xs:attribute name="RemotePort" type="xs:long" default="0"/>

attribute **UdpClientAppType/@PacketSize**

type	xs:long
properties	default 100
source	<xs:attribute name="PacketSize" type="xs:long" default="100"/>

complexType **UdpServerAppType**

diagram	<pre> classDiagram class UdpServerAppType class AbstractApplicationType { <<extension>> <<attribute Name>> <<attribute Start>> <<attribute Stop>> <<attribute Port>> <<attribute PacketWindowSize>> } UdpServerAppType < -- AbstractApplicationType </pre>																																				
type	extension of AbstractApplicationType																																				
properties	base AbstractApplicationType																																				
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Start</td> <td>xs:double</td> <td>optional</td> <td>1.0</td> <td></td> <td></td> </tr> <tr> <td>Stop</td> <td>xs:double</td> <td>optional</td> <td>10.0</td> <td></td> <td></td> </tr> <tr> <td>Port</td> <td>xs:long</td> <td></td> <td>9</td> <td></td> <td></td> </tr> <tr> <td>PacketWindowSize</td> <td>xs:long</td> <td></td> <td>32</td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	xs:string	required				Start	xs:double	optional	1.0			Stop	xs:double	optional	10.0			Port	xs:long		9			PacketWindowSize	xs:long		32		
Name	Type	Use	Default	Fixed	Annotation																																
Name	xs:string	required																																			
Start	xs:double	optional	1.0																																		
Stop	xs:double	optional	10.0																																		
Port	xs:long		9																																		
PacketWindowSize	xs:long		32																																		
source	<pre> <xs:complexType name="UdpServerAppType"> <xs:complexContent> <xs:extension base="AbstractApplicationType"> <xs:attribute name="Port" type="xs:long" default="9"/> <xs:attribute name="PacketWindowSize" type="xs:long" default="32"/> </xs:extension> <!-- Port on which we listen for incoming packets.. --> </xs:complexContent> </xs:complexType> </pre>																																				

	<code></xs:complexType></code>
--	--------------------------------------

attribute **UdpServerAppType/@Port**

type	xs:long
properties	default 9
source	<code><xs:attribute name="Port" type="xs:long" default="9"/></code>

attribute **UdpServerAppType/@PacketWindowSize**

type	xs:long
properties	default 32
source	<code><xs:attribute name="PacketWindowSize" type="xs:long" default="32"/></code>

complexType **UdpEchoClientAppType**

diagram	<pre> classDiagram class AbstractApplicationType { <<AbstractApplicationType (extension)>> attributes Name Start Stop } class UdpEchoClientAppType { <<UdpEchoClientAppType>> attributes MaxPackets Interval RemoteAddress RemotePort PacketSize } UdpEchoClientAppType < -- AbstractApplicationType </pre>																																				
type	extension of AbstractApplicationType																																				
properties	base <code>AbstractApplicationType</code>																																				
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Start</td> <td>xs:double</td> <td>optional</td> <td>1.0</td> <td></td> <td></td> </tr> <tr> <td>Stop</td> <td>xs:double</td> <td>optional</td> <td>10.0</td> <td></td> <td></td> </tr> <tr> <td>MaxPackets</td> <td>xs:long</td> <td></td> <td>100</td> <td></td> <td></td> </tr> <tr> <td>Interval</td> <td>xs:double</td> <td></td> <td>1.0</td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	xs:string	required				Start	xs:double	optional	1.0			Stop	xs:double	optional	10.0			MaxPackets	xs:long		100			Interval	xs:double		1.0		
Name	Type	Use	Default	Fixed	Annotation																																
Name	xs:string	required																																			
Start	xs:double	optional	1.0																																		
Stop	xs:double	optional	10.0																																		
MaxPackets	xs:long		100																																		
Interval	xs:double		1.0																																		

	<p><u>RemoteAddress</u> IPv4AddressType required</p> <p><u>RemotePort</u> xs:long 0</p> <p><u>PacketSize</u> xs:long 100</p>
source	<pre><xs:complexType name="UdpEchoClientAppType"> <xs:complexContent> <xs:extension base="AbstractApplicationType"> <xs:attribute name="MaxPackets" type="xs:long" default="100"/> <xs:attribute name="Interval" type="xs:double" default="1.0"/> <xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/> <xs:attribute name="RemotePort" type="xs:long" default="0"/> <xs:attribute name="PacketSize" type="xs:long" default="100"/> </xs:extension> <!-- The maximum number of packets the application will send --&gt; <!-- The time to wait between packets --&gt; <!-- Address of remote destination . --&gt; <!--The destination Port of the outbound packets. --&gt; <!-- Size of echo data in outbound packets --&gt; &lt;/xs:complexContent&gt; &lt;/xs:complexType&gt;</pre> </pre>

attribute UdpEchoClientAppType/@MaxPackets

type	xs:long
properties	default 100
source	<pre><xs:attribute name="MaxPackets" type="xs:long" default="100"/></pre>

attribute UdpEchoClientAppType/@Interval

type	xs:double
properties	default 1.0
source	<pre><xs:attribute name="Interval" type="xs:double" default="1.0"/></pre>

attribute UdpEchoClientAppType/@RemoteAddress

type	IPv4AddressType						
properties	use required						
facets	<table> <tr> <td>Kind</td> <td>Value</td> <td>Annotation</td> </tr> <tr> <td>pattern</td> <td>((25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[0-9][0-9]) (25[0-5] 2[0-4][0-9] 1[0-9][0-9]) (1[0-9][0-9])){3}((25[0-5] 2[0-4][0-9] 1[0-9][0-9]) (1[0-9][0-9])){3}</td> <td></td> </tr> </table>	Kind	Value	Annotation	pattern	((25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[0-9][0-9]) (25[0-5] 2[0-4][0-9] 1[0-9][0-9]) (1[0-9][0-9])){3}((25[0-5] 2[0-4][0-9] 1[0-9][0-9]) (1[0-9][0-9])){3}	
Kind	Value	Annotation					
pattern	((25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[0-9][0-9]) (25[0-5] 2[0-4][0-9] 1[0-9][0-9]) (1[0-9][0-9])){3}((25[0-5] 2[0-4][0-9] 1[0-9][0-9]) (1[0-9][0-9])){3}						
source	<pre><xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/></pre>						

attribute UdpEchoClientAppType/@RemotePort

type	xs:long
properties	default 0
source	<pre><xs:attribute name="RemotePort" type="xs:long" default="0"/></pre>

attribute **UdpEchoClientAppType/@PacketSize**

type	xs:long
properties	default 100
source	<xs:attribute name="PacketSize" type="xs:long" default="100"/>

complexType **UdpEchoServerAppType**

diagram	<pre> classDiagram class UdpEchoServerAppType { <<extension of AbstractApplicationType>> <<attributes>> Name Start Stop Port } </pre>																														
type	extension of AbstractApplicationType																														
properties	base AbstractApplicationType																														
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Start</td> <td>xs:double</td> <td>optional</td> <td>1.0</td> <td></td> <td></td> </tr> <tr> <td>Stop</td> <td>xs:double</td> <td>optional</td> <td>10.0</td> <td></td> <td></td> </tr> <tr> <td>Port</td> <td>xs:long</td> <td></td> <td>9</td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	xs:string	required				Start	xs:double	optional	1.0			Stop	xs:double	optional	10.0			Port	xs:long		9		
Name	Type	Use	Default	Fixed	Annotation																										
Name	xs:string	required																													
Start	xs:double	optional	1.0																												
Stop	xs:double	optional	10.0																												
Port	xs:long		9																												
source	<pre> <xs:complexType name="UdpEchoServerAppType"> <xs:complexContent> <xs:extension base="AbstractApplicationType"> <xs:attribute name="Port" type="xs:long" default="9"/> </xs:extension> <!-- Port on which we listen for incoming packets.. --> </xs:complexContent> </xs:complexType> </pre>																														

attribute **UdpEchoServerAppType/@Port**

type	xs:long
properties	default 9
source	<xs:attribute name="Port" type="xs:long" default="9"/>

complexType **UdpTraceClientAppType**

diagram	<pre> classDiagram class UdpTraceClientAppType class AbstractApplicationType { <<extension>> <<attributes>> Name Start Stop MaxPacketSize RemoteAddress RemotePort } UdpTraceClientAppType < -- AbstractApplicationType </pre>																																										
type	extension of AbstractApplicationType																																										
properties	base AbstractApplicationType																																										
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Start</td> <td>xs:double</td> <td>optional</td> <td>1.0</td> <td></td> <td></td> </tr> <tr> <td>Stop</td> <td>xs:double</td> <td>optional</td> <td>10.0</td> <td></td> <td></td> </tr> <tr> <td>MaxPacketSize</td> <td>xs:long</td> <td></td> <td>1024</td> <td></td> <td></td> </tr> <tr> <td>RemoteAddress</td> <td>IPv4AddressType</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>RemotePort</td> <td>xs:long</td> <td></td> <td>100</td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	xs:string	required				Start	xs:double	optional	1.0			Stop	xs:double	optional	10.0			MaxPacketSize	xs:long		1024			RemoteAddress	IPv4AddressType	required				RemotePort	xs:long		100		
Name	Type	Use	Default	Fixed	Annotation																																						
Name	xs:string	required																																									
Start	xs:double	optional	1.0																																								
Stop	xs:double	optional	10.0																																								
MaxPacketSize	xs:long		1024																																								
RemoteAddress	IPv4AddressType	required																																									
RemotePort	xs:long		100																																								
source	<pre> <xs:complexType name="UdpTraceClientAppType"> <xs:complexContent> <xs:extension base="AbstractApplicationType"> <xs:attribute name="MaxPacketSize" type="xs:long" default="1024"/> <xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/> <xs:attribute name="RemotePort" type="xs:long" default="100"/> </xs:extension> <!-- The maximum size of a packet. --> <!-- Address of remote destination . --> <!-- The remote destination Port of the outbound packets. --> </xs:complexContent> </xs:complexType> </pre>																																										

attribute **UdpTraceClientAppType/@MaxPacketSize**

type	xs:long
properties	default 1024
source	<pre> <xs:attribute name="MaxPacketSize" type="xs:long" default="1024"/> </pre>

attribute **UdpTraceClientAppType/@RemoteAddress**

type	IPv4AddressType
properties	use required
facets	Kind Value pattern <code>((25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[1-9][0-9]) ([0-9]).{3}(25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[1-9][0-9]) ([0-9])</code> Annotation
source	<code><xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/></code>

attribute **UdpTraceClientAppType/@RemotePort**

type	xs:long
properties	default 100
source	<code><xs:attribute name="RemotePort" type="xs:long" default="100"/></code>

complexType **V4PingType**

diagram	<pre> classDiagram class AbstractApplicationType { <<AbstractApplicationType (extension)>> <<attributes>> Name Start Stop } class V4PingType { <<attributes>> Interval Size Verbose RemoteAddress } V4PingType < -- AbstractApplicationType </pre>																																																
type	extension of AbstractApplicationType																																																
properties	base AbstractApplicationType																																																
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Start</td> <td>xs:double</td> <td>optional</td> <td>1.0</td> <td></td> <td></td> </tr> <tr> <td>Stop</td> <td>xs:double</td> <td>optional</td> <td>10.0</td> <td></td> <td></td> </tr> <tr> <td>Interval</td> <td>xs:long</td> <td></td> <td>1</td> <td></td> <td></td> </tr> <tr> <td>Size</td> <td>xs:long</td> <td></td> <td>56</td> <td></td> <td></td> </tr> <tr> <td>Verbose</td> <td>xs:boolean</td> <td></td> <td>false</td> <td></td> <td></td> </tr> <tr> <td>RemoteAddress</td> <td>IPv4AddressType</td> <td>required</td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	xs:string	required				Start	xs:double	optional	1.0			Stop	xs:double	optional	10.0			Interval	xs:long		1			Size	xs:long		56			Verbose	xs:boolean		false			RemoteAddress	IPv4AddressType	required			
Name	Type	Use	Default	Fixed	Annotation																																												
Name	xs:string	required																																															
Start	xs:double	optional	1.0																																														
Stop	xs:double	optional	10.0																																														
Interval	xs:long		1																																														
Size	xs:long		56																																														
Verbose	xs:boolean		false																																														
RemoteAddress	IPv4AddressType	required																																															

source	<pre> <xs:complexType name="V4PingType"> <xs:complexContent> <xs:extension base="AbstractApplicationType"> <xs:attribute name="Interval" type="xs:long" default="1"/> <xs:attribute name="Size" type="xs:long" default="56"/> <xs:attribute name="Verbose" type="xs:boolean" default="false"/> <xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/> </xs:extension> <!-- Wait interval seconds between sending each packet. --> <!-- Size of echo data in outbound packets --> <!-- Produce usual output. --> <!-- Address of remote destination . --> </xs:complexContent> </xs:complexType> </pre>
--------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

attribute V4PingType/@Interval

type	xs:long
properties	default 1
source	<xs:attribute name="Interval" type="xs:long" default="1"/>

attribute V4PingType/@Size

type	xs:long
properties	default 56
source	<xs:attribute name="Size" type="xs:long" default="56"/>

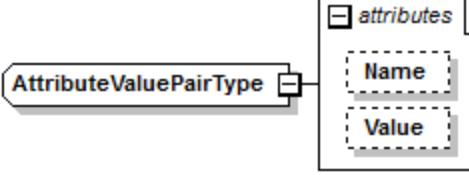
attribute V4PingType/@Verbose

type	xs:boolean
properties	default false
source	<xs:attribute name="Verbose" type="xs:boolean" default="false"/>

attribute V4PingType/@RemoteAddress

type	IPv4AddressType						
properties	use required						
facets	<table> <tr> <td>Kind</td> <td>Value</td> <td>Annotation</td> </tr> <tr> <td>pattern</td> <td>((25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[1-9][0-9][0-9])\.).{3}(25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[1-9][0-9][0-9])</td> <td></td> </tr> </table>	Kind	Value	Annotation	pattern	((25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[1-9][0-9][0-9])\.).{3}(25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[1-9][0-9][0-9])	
Kind	Value	Annotation					
pattern	((25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[1-9][0-9][0-9])\.).{3}(25[0-5] 2[0-4][0-9] 1[0-9][0-9] 1[1-9][0-9][0-9])						
source	<xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/>						

complexType **AttributeValuePairType**

diagram																			
used by	element GenericAppType/Attribute																		
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td><code>xs:string</code></td> <td></td> <td></td> <td></td> <td></td> </tr> <tr> <td>Value</td> <td><code>xs:string</code></td> <td></td> <td></td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	<code>xs:string</code>					Value	<code>xs:string</code>				
Name	Type	Use	Default	Fixed	Annotation														
Name	<code>xs:string</code>																		
Value	<code>xs:string</code>																		
source	<pre><xs:complexType name="AttributeValuePairType"> <xs:attribute name="Name" type="xs:string"/> <xs:attribute name="Value" type="xs:string"/> </xs:complexType></pre>																		

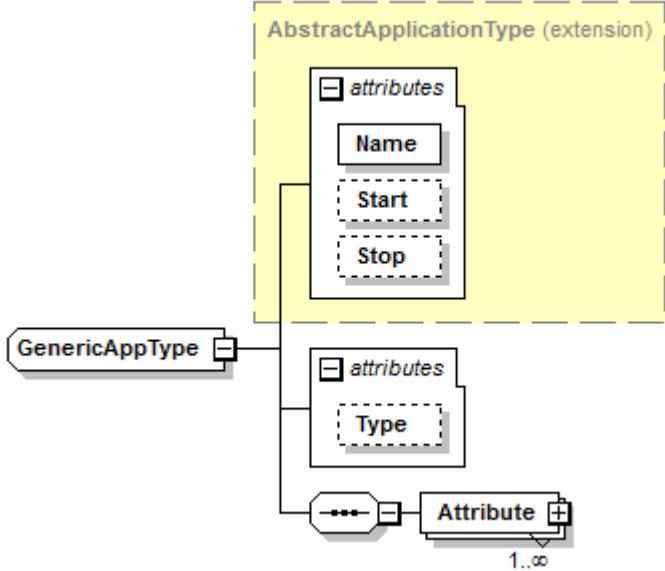
attribute **AttributeValuePairType/@Name**

type	<code>xs:string</code>
source	<pre><xs:attribute name="Name" type="xs:string"/></pre>

attribute **AttributeValuePairType/@Value**

type	<code>xs:string</code>
source	<pre><xs:attribute name="Value" type="xs:string"/></pre>

complexType **GenericAppType**

diagram	
type	extension of AbstractApplicationType

properties	base AbstractApplicationType					
children	Attribute					
attributes	Name Name	Type xs:string	Use required	Default	Fixed	Annotation
	Start	xs:double	optional	1.0		
	Stop	xs:double	optional	10.0		
	Type	xs:string				
source	<pre><xs:complexType name="GenericAppType"> <xs:complexContent> <xs:extension base="AbstractApplicationType"> <xs:sequence> <xs:element name="Attribute" type="AttributeValuePairType" maxOccurs="unbounded"/> </xs:sequence> <xs:attribute name="Type" type="xs:string"/> </xs:extension> </xs:complexContent> </xs:complexType></pre>					

attribute **GenericAppType/@Type**

type	xs:string
source	<pre><xs:attribute name="Type" type="xs:string"/></pre>

element **GenericAppType/Attribute**

diagram	<pre> classDiagram class AttributeValuePairType { attributes +--> Attribute } class Attribute { <<1..>> } class Name class Value </pre>					
type	AttributeValuePairType					
properties	minOcc 1 maxOcc unbounded content complex					
attributes	Name Name	Type xs:string	Use	Default	Fixed	Annotation
	Value	xs:string				
source	<pre><xs:element name="Attribute" type="AttributeValuePairType" maxOccurs="unbounded"/></pre>					

complexType Simple WebClientAppType

diagram	<pre> classDiagram class SimpleWebClientAppType class AbstractApplicationType { <<extension>> <<attributes>> Name Start Stop } SimpleWebClientAppType --> AbstractApplicationType <<attributes>> MaxPackets Interval RemoteAddress RemotePort PacketSize OffTime MaxBytes </pre>																																																																		
type	extension of AbstractApplicationType																																																																		
properties	base AbstractApplicationType																																																																		
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Start</td> <td>xs:double</td> <td>optional</td> <td>1.0</td> <td></td> <td></td> </tr> <tr> <td>Stop</td> <td>xs:double</td> <td>optional</td> <td>10.0</td> <td></td> <td></td> </tr> <tr> <td>MaxPackets</td> <td>xs:long</td> <td></td> <td>100</td> <td></td> <td></td> </tr> <tr> <td>Interval</td> <td>xs:double</td> <td></td> <td>1.0</td> <td></td> <td></td> </tr> <tr> <td>RemoteAddress</td> <td>IPv4AddressType</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>RemotePort</td> <td>xs:long</td> <td></td> <td>0</td> <td></td> <td></td> </tr> <tr> <td>PacketSize</td> <td>xs:long</td> <td></td> <td>512</td> <td></td> <td></td> </tr> <tr> <td>OffTime</td> <td>xs:double</td> <td></td> <td>100.0</td> <td></td> <td></td> </tr> <tr> <td>MaxBytes</td> <td>xs:long</td> <td></td> <td>0</td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	xs:string	required				Start	xs:double	optional	1.0			Stop	xs:double	optional	10.0			MaxPackets	xs:long		100			Interval	xs:double		1.0			RemoteAddress	IPv4AddressType	required				RemotePort	xs:long		0			PacketSize	xs:long		512			OffTime	xs:double		100.0			MaxBytes	xs:long		0		
Name	Type	Use	Default	Fixed	Annotation																																																														
Name	xs:string	required																																																																	
Start	xs:double	optional	1.0																																																																
Stop	xs:double	optional	10.0																																																																
MaxPackets	xs:long		100																																																																
Interval	xs:double		1.0																																																																
RemoteAddress	IPv4AddressType	required																																																																	
RemotePort	xs:long		0																																																																
PacketSize	xs:long		512																																																																
OffTime	xs:double		100.0																																																																
MaxBytes	xs:long		0																																																																
source	<pre> <xs:complexType name="SimpleWebClientAppType"> <xs:complexContent> <xs:extension base="AbstractApplicationType"> <xs:attribute name="MaxPackets" type="xs:long" default="100"/> <xs:attribute name="Interval" type="xs:double" default="1.0"/> <xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/> <xs:attribute name="RemotePort" type="xs:long" default="0"/> <xs:attribute name="PacketSize" type="xs:long" default="512"/> <xs:attribute name="OffTime" type="xs:double" default="100.0"/> <xs:attribute name="MaxBytes" type="xs:long" default="0"/> </xs:extension> </xs:complexContent> </xs:complexType> </pre>																																																																		

	<pre> </xs:extension> <!-- The maximum number of packets the application will send --> <!-- The time to wait between packets --> <!-- Address of remote destination . --> <!--The destination Port of the outbound packets. --> <!-- Size of echo data in outbound packets --> <!-- Off rate --> <!-- Max bytes to send --> </xs:complexContent> </xs:complexType> </pre>
--	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

attribute Simple WebClientAppType/@MaxPackets

type	xs:long
properties	default 100
source	<pre><xs:attribute name="MaxPackets" type="xs:long" default="100"/></pre>

attribute Simple WebClientAppType/@Interval

type	xs:double
properties	default 1.0
source	<pre><xs:attribute name="Interval" type="xs:double" default="1.0"/></pre>

attribute Simple WebClientAppType/@RemoteAddress

type	IPv4AddressType						
properties	use required						
facets	<table> <thead> <tr> <th>Kind</th> <th>Value</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>pattern</td> <td>((25[0-5] 2[0-4][0-9] 1[0-9][0-9])[1-9][0-9]) (3){(25[0-5] 2[0-4][0-9] 1[0-9][0-9]) (1-9)[0-9]) (0-9))</td> <td></td> </tr> </tbody> </table>	Kind	Value	Annotation	pattern	((25[0-5] 2[0-4][0-9] 1[0-9][0-9])[1-9][0-9]) (3){(25[0-5] 2[0-4][0-9] 1[0-9][0-9]) (1-9)[0-9]) (0-9))	
Kind	Value	Annotation					
pattern	((25[0-5] 2[0-4][0-9] 1[0-9][0-9])[1-9][0-9]) (3){(25[0-5] 2[0-4][0-9] 1[0-9][0-9]) (1-9)[0-9]) (0-9))						
source	<pre><xs:attribute name="RemoteAddress" type="IPv4AddressType" use="required"/></pre>						

attribute Simple WebClientAppType/@RemotePort

type	xs:long
properties	default 0
source	<pre><xs:attribute name="RemotePort" type="xs:long" default="0"/></pre>

attribute Simple WebClientAppType/@PacketSize

type	xs:long
properties	default 512
source	<pre><xs:attribute name="PacketSize" type="xs:long" default="512"/></pre>

attribute **SimpleWebClientAppType/@OffTime**

type	xs:double
properties	default 100.0
source	<xs:attribute name="OffTime" type="xs:double" default="100.0"/>

attribute **SimpleWebClientAppType/@MaxBytes**

type	xs:long
properties	default 0
source	<xs:attribute name="MaxBytes" type="xs:long" default="0"/>

complexType **SimpleWebServerAppType**

diagram	<pre> classDiagram class SimpleWebServerAppType class AbstractApplicationType { <<extension point>> attribute Name attribute Start attribute Stop } SimpleWebServerAppType < -- AbstractApplicationType <<extension point>> { attribute Port } </pre>																														
type	extension of AbstractApplicationType																														
properties	base AbstractApplicationType																														
attributes	<table> <thead> <tr> <th>Name</th> <th>Type</th> <th>Use</th> <th>Default</th> <th>Fixed</th> <th>Annotation</th> </tr> </thead> <tbody> <tr> <td>Name</td> <td>xs:string</td> <td>required</td> <td></td> <td></td> <td></td> </tr> <tr> <td>Start</td> <td>xs:double</td> <td>optional</td> <td>1.0</td> <td></td> <td></td> </tr> <tr> <td>Stop</td> <td>xs:double</td> <td>optional</td> <td>10.0</td> <td></td> <td></td> </tr> <tr> <td>Port</td> <td>xs:long</td> <td></td> <td>9</td> <td></td> <td></td> </tr> </tbody> </table>	Name	Type	Use	Default	Fixed	Annotation	Name	xs:string	required				Start	xs:double	optional	1.0			Stop	xs:double	optional	10.0			Port	xs:long		9		
Name	Type	Use	Default	Fixed	Annotation																										
Name	xs:string	required																													
Start	xs:double	optional	1.0																												
Stop	xs:double	optional	10.0																												
Port	xs:long		9																												
source	<pre> <xs:complexType name="SimpleWebServerAppType"> <xs:complexContent> <xs:extension base="AbstractApplicationType"> <xs:attribute name="Port" type="xs:long" default="9"/> </xs:extension> <!-- Port on which we listen for incoming packets.. --> </xs:complexContent> </xs:complexType> </pre>																														

attribute SimpleWebServerAppType/@Port

type	xs:long
properties	default 9
source	<xs:attribute name="Port" type="xs:long" default="9"/>

XML Schema documentation generated by [XMLSpy](#) Schema Editor <http://www.altova.com/xmlspy>